



Горячее Лучшее Свежее ...



Войти



emprenoso 3 часа назад



Искусственный интеллект

AI на кухне: как я заставил нейросеть искать хорошие комментарии про себя

История о том, как я хотел провести анализ комментариев, а в итоге получил неожиданный, но полезный опыт с локальным AI.

	A	B	C	D	E	F	G
1	Ресурс	Кол-во	Начало	Конец	Тираж (печатн.)	Просмотры (веб)	Комментарии (веб)
2	https://habr.com	113	2019-11	2025-09	0,0	2 192 876,0	4 497,0
3	https://pikabu.ru	61	2024-09	2025-09	0,0	506 243,0	726,0
4	https://smart-lab.ru	50	2024-09	2025-09	0,0	453 955,0	1 998,0
5	https://www.youtube.com	22	2013-02	2025-09	0,0	182 381,0	22,0
6	https://alenka.capital	1	2025-09	2025-09	0,0		0,0
7	https://fkviking.timepad.ru	1	2025-09	2025-09	0,0		0,0
8	https://rationalanswer.club	7	2024-11	2025-08	0,0	45 078,0	39,0
9	https://vc.ru	11	2019-11	2025-08	0,0	11 447,0	9,0
10	https://t.me	12	2019-03	2024-12	0,0	79 200,0	122,0
11	https://www.1tv.ru	1	2024-12	2024-12	0,0		0,0
12	https://sprut.ai	3	2024-05	2024-11	0,0		0,0
13	https://t-j.ru	49	2020-04	2024-10	0,0	2 300 823,0	4 186,0
14	https://github.com	10	2019-09	2023-06	0,0	0,0	0,0
15	https://podcast.ru	1	2022-11	2022-11	0,0		0,0
16	https://medium.com	1	2020-02	2020-02	0,0		0,0
17	https://zen.yandex.ru	1	2019-12	2019-12	0,0		0,0
18	https://3dtoday.ru	13	2018-09	2018-11	0,0	432 787,0	553,0
19	Журнал Инновации и инвестиции	2	2018-04	2018-05	600,0		0,0
20	Сборник ПНИПУ	7	2012-04	2017-05	700,0		0,0
21	https://z-wave.ru	16	2014-11	2015-04	0,0		0,0

LynxReport: учёт публикаций [Node.js]

Недавно передо мной встала задача собрать все положительные комментарии к моим статьям. Веду их учёт в таблице, и там уже вполне серьёзные цифры — больше 300 строк и свыше 10 тысяч комментариев. Основные площадки, где я публикуюсь, выглядят так:

- Хабр — 4 497 комментариев
- Т-Ж — 4 186
- Смартлаб — 1 998
- Пикабу — 726

Вручную искать в этом массиве текста слова поддержки — долго и нудно, а главное — совершенно не масштабируется. Так родилась идея: поручить всё локальной нейросети для анализа тональности. Заодно я хотел на практике разобраться с моделями на основе BERT.

Для этой цели у меня был, как мне казалось, идеальный инструмент: компьютер с Ubuntu 24 LTS, Docker и мощной NVIDIA GeForce RTX 5060 Ti на 16 ГБ. Казалось, что с RTX 5060 Ti на 16 ГБ анализ пройдёт молниеносно. Но на деле именно GPU стал главной точкой отказа — пришлось всё считать на CPU.

Код на GitHub.

Войти

Логин

Пароль

Войти

Создать аккаунт

[Забыли пароль?](#)

или продолжите с



Войти с Яндекс ID



Войти через VK ID



Промокоды



Работа



Курсы



Реклама

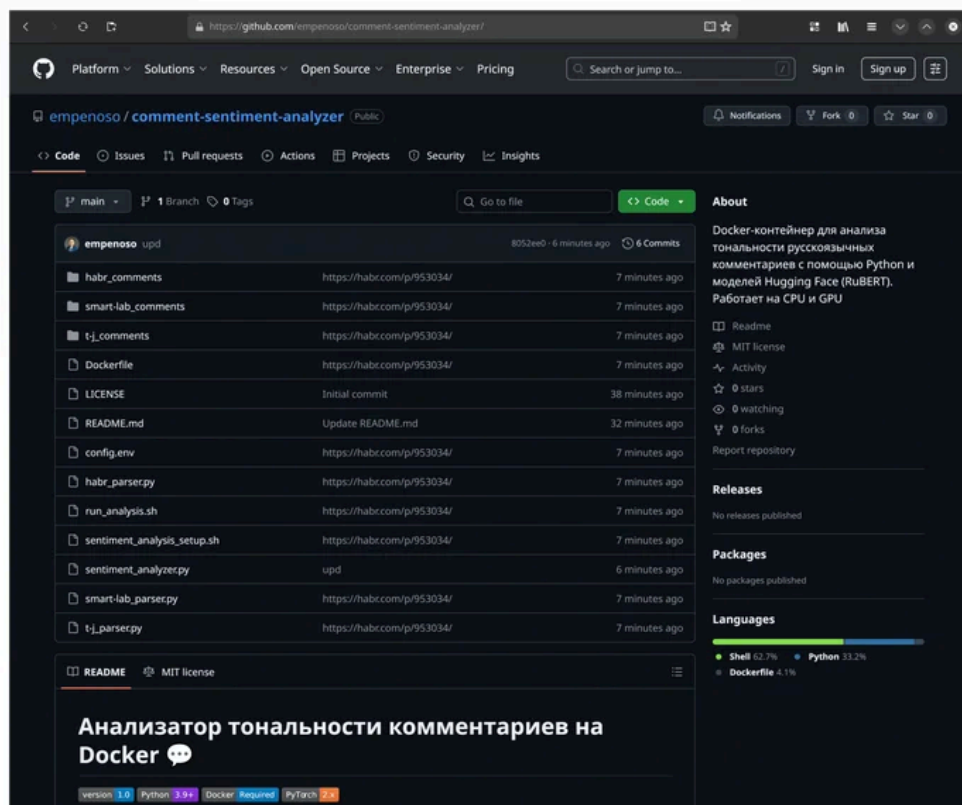


Игры



Пополнение Steam

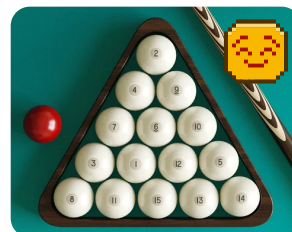
Что делает мой скрипт на GitHub



<https://github.com/empenoso/comment-sentiment-analyzer/>

Прежде чем бросаться в бой с CUDA, нужно было подготовить данные. Комментарии разбросаны по разным сайтам без возможности экспорта, поэтому я написал несколько парсеров, которые собирают тексты в единый JSON-формат — один файл на статью.

Пикабу Игры +1000 бесплатных онлайн игр



Бильярд 3D: Русский бильярд

Симуляторы, Спорт,
Настольные

Играть

Топ прошлой недели

- charlesrus
7 постов
- Animalrescued
55 постов
- Denk13
6 постов

[Посмотреть весь топ](#)



Лучшие посты
недели



JSON Хабр

Парсеры я оставил приватными, так как они заточены под мои задачи, но [ядро системы выложил на GitHub](#).

Скрипт [sentiment_analyzer.py](#) берёт JSON-файлы, подготовленные на первом этапе, и пропускает текст каждого комментария через предварительно обученную нейросетевую модель [cointegrated/rubert-tiny-sentiment-balanced](#). Модель определяет эмоциональную окраску текста — позитивную, негативную или нейтральную.

Скрипт задуман как универсальный инструмент, поэтому основные параметры вынесены в `config.env`.

- `MODEL_NAME`: можно указать любую другую модель с Hugging Face.
- `DEVICE`: позволяет выбрать, на чём производить вычисления — на `cpu` или `cuda` (GPU).
- `POSITIVE_THRESHOLD`: порог уверенности модели, чтобы отнести комментарий к позитивным.
- `EXCLUDE_AUTHORS`: список авторов, чьи комментарии нужно игнорировать (например, мои собственные ответы).
- `MIN_COMMENT_LENGTH`: отсеивает слишком короткие и неинформативные комментарии.

Оркестрация с помощью Docker и Shell

Для удобства запуска и настройки я написал два `shell`-скрипта:

Рассылка Пикабу:
отправляем самые
рейтинговые материалы за 7
дней 🔥

Укажи [Подписаться](#)

Нажимая «Подписаться»,
я даю согласие на [обработку
данных и условия почтовых
рассылок](#).

Помощь	Правила
Кодекс Пикабу	соцсети
Команда	О
Пикабу	рекомендация
Моб.	х
приложение	О компании

Промокоды Биг Гик
Промокоды Lamoda
Промокоды МВидео
Промокоды Яндекс Маркет
Промокоды Отелло
Промокоды Aroma Butik
Промокоды Яндекс
Путешествия
Постила
Футбол сегодня



1. [sentiment_analysis_setup.sh](#) (это не ссылка, Пикабу так размечает): этот скрипт — для первоначальной настройки. Он проверяет систему, устанавливает Docker и NVIDIA Container Toolkit, создаёт необходимые папки и конфигурационный файл. Запустив его один раз, вы подготавливаете окружение для дальнейшей работы.
2. [run_analysis.sh](#) (это не ссылка, Пикабу так размечает): простой скрипт для запуска анализа. Он читает конфигурацию из `config.env` и запускает Docker-контейнер с нужными параметрами.

На практике это сводится к трём шагам: подготовка системы через `setup`-скрипт, сбор комментариев парсерами и запуск анализа через [run_analysis.sh](#).

Все найденные позитивные комментарии скрипт аккуратно складывает в текстовые файлы.

Успех на CPU и урок о масштабировании

После череды падений с CUDA и финальной ошибки «No kernel image» пришлось смириться: GPU в проекте не будет. Я открыл `config.env`, поменял `DEVICE=cuda` на `DEVICE=cpu` и нажал `save`.

RuBERT-tiny справился за полторы минуты — все 10 000 комментариев были разобраны. Вся похвала и поддержка оказались в аккуратных текстовых файлах.

На финишной прямой я переписал логику сохранения. Вместо сотен мелких JSON теперь формируется один аккуратный текстовый файл для каждой площадки.

Структура вывода проста:

Автор: Андрей Мищенко

Дата: 2025-06-06T11:24:20.551316+03:00

Текст: Полезные формулы, спасибо!

Ссылка: <https://t-j.ru/guide/excel-kotirovki/#c2857460>

Автор: Whalerman

Дата: 2025-09-09T07:40:00.000Z

Текст: Михаил, спасибо! Хорошие и полезные посты!

Ссылка: <https://smart-lab.ru/blog/1202442.php#comment18571129>

Автор: DashBerlin

Дата: 2025-08-23T00:18:43.000Z

Текст: Впервые решил заглянуть в подобный обзор, порадовала позиция количество закладок, интересно, после эти публикации читаются. Я подписан на автора, он пачками сохраняет статьи в закладки, какой процент он потом перечитывает из этого. Спасибо за обзор))

Ссылка: https://habr.com/ru/articles/933806/comments/#comment_28742672

Каждый комментарий — отдельный блок, удобно читать и анализировать.

Я ожидал, что GPU будет обрабатывать тысячи комментариев в секунду, но реальность оказалась прагматичнее — CPU выдал скорость около 110

комментариев в секунду.

Этот опыт наглядно показал: локальный AI на процессоре — отличное решение для задач исследователя-одиночки. Но если бы у меня был миллиард строк, этот подход бы провалился. Это инструмент для прототипирования и персональных проектов, а не для Big Data.

Заключение: главный урок для AI-энтузиастов

Эта история с видеокартой преподавала мне два важных урока.

Во-первых, самое новое оборудование — не всегда самое лучшее. Моя RTX 5060 Ti с передовой архитектурой Blackwell оказалась настолько свежей, что стабильный PyTorch просто не умел с ней работать. В погоне за технологиями легко обогнать экосистему и остаться с мощным, но бесполезным инструментом (я знаю [про обходной путь](#)). Иногда проверенная карта предыдущего поколения — более разумный выбор.

Во-вторых, Docker — это не просто среда для запуска, а настоящая страховка. Он позволил мне безболезненно переключаться между конфигурациями и быстро откатиться на CPU, когда GPU подвёл. Именно изоляция в контейнере спасла проект, позволив проиграть «битву за CUDA», но всё равно выиграть войну.

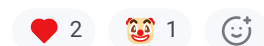
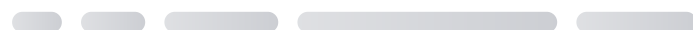
В итоге, мой проект заработал не благодаря дорогой видеокарте, а вопреки ей. Процессор решил задачу быстрее, чем я успел допить чай. Это доказывает, что локальный AI — не удел облачных гигантов. Он вполне доступен на обычных ПК, если подходить к делу прагматично и помнить, что иногда самое простое решение — самое верное.

Автор: Михаил Шардин

[🔗 Моя онлайн-визитка](#)

[📱 Telegram «Умный Дом Инвестора»](#)

7 октября 2025



**Искусственный интеллект**

4.7К поста • 11.3К подписчиков

[Добавить пост](#)[Подписаться](#)**Правила сообщества**

ВНИМАНИЕ! В сообществе запрещена публикация генеративного контента без детального описания промтов и процесса получения публикуемого результата.

...

[Подробнее](#) ✓

Чтобы оставить комментарий, необходимо [зарегистрироваться](#) или [войти](#)

● — ■ —

—
—
—
—
—
—

● — ■ —

—
—
—
—
—
—

● — ■ —

—
—
—
—
—
—

