

Во время посещения сайта вы соглашаетесь с использованием файлов [cookie](#)

[Хорошо](#)



Михаил Шардин ★

личный блог



23 сентября 2025, 04:56

[+ Подписаться](#)

Умный диктофон: почему лёгких решений не бывает

Несколько недель назад я опубликовал статью о том, [как превратить обычный диктофон в инструмент для расшифровки речи с помощью OpenAI Whisper](#). Идея была создать бесплатную и приватную систему ИИ диктофона, которая избавляет от необходимости переслушивать аудиозаписи лекций или выступлений.



Plaud note pin

ИИ-помощник сделает транскрипцию всей аудио и составит реsume

Компактный размер

13+ шаблонов для профессиональных отзывов

до 20 часов непрерывной записи

850 баллов за отзыв

Новинка

18126 ₽

Умный диктофон Plaud Note Pin (Cosmic gray) с чатом GPT - диктофон с искусственным...

★ 4.9 ● 158 отзывов

18 сентября



СХЕМАТЕХ ПРОФЕССИОНАЛЬНЫЙ ДИКТОФОН KP-666

16 ГБ

32 ГБ

Новинка

7990 ₽

Профессиональный цифровой диктофон KP 666 с 16 гб...

★ 4.9 ● 90 отзывов

20 сентября



Диктофон профессиональный

100ч время записи

32ГБ объем памяти

активация голосом

Распродажа

3179 ₽ 9990 ₽ -68%

61шт осталось

СХЕМАТЕХ

Диктофон профессиональный V65 / MP3-плеер / 32GB

★ 4.9 ● 90 отзывов

20 сентября



ДИКТОФОН

с искусственным интеллектом

- расшифровка записи
- преобразует звук в текст

PLAUD

ChatGPT-4

активация голосом

Распродажа

19 301 ₽ 59 990 ₽ -67%

70шт осталось

СХЕМАТЕХ

Диктофон PLAUD с искусственным интеллектом...

★ 4.9 ● 90 отзывов

20 сентября

ИИ и обычные диктофоны

В процессе настройки я боролся с несовместимостью библиотек, подбирал нужные версии драйверов и вручную собирали рабочее окружение. В комментариях мне справедливо заметили: «Вместо всей этой возни можно было найти готовый Docker-контейнер и поднять всё одной командой». Звучало логично, и я с энтузиазмом принял этот совет. *Я ведь верю людям в интернете.*

Новая идея — не просто расшифровывать речь, а разделять её по голосам — как на совещании или встрече. Это называется диаризацией, и для неё существует продвинутая

версия — WhisperX. Цель была проста — получить на выходе не сплошное полотно, а

Введите текст комментария

Что я услыхал... Путь «в одну команду» оказался полон сюрпризов — все свалилось одновременно: то скрипт не видел мои файлы, то не мог получить к ним доступ, то просто зависал без объяснения причин. Внутри этой «волшебной упаковки» царил хаос, и мне приходилось разбираться, почему она не хочет работать. Я спросил у Алисы в Поиске, как оптимизировать работу WhisperX с диаризацией. Она предложила несколько рекомендаций по настройке параметров обработки, которые помогли улучшить качество распознавания:

Новый в чате Алиса

На основе источников, возможны неточности

Процесс поиска ▾

Оптимизировать работу WhisperX с диаризацией (определением говорящих) можно, настроив параметры, выбрав подходящую модель, используя параллельную обработку или интегрируя WhisperX с другими системами. WhisperX расширяет возможности оригинального Whisper, обеспечивая точную разбивку текста по времени и определение говорящих, и диаризация позволяет идентифицировать разных говорящих в аудио.

[aigenom.ru](#) [modal.com](#) [dev.to](#)

Настройка параметров

- Минимизировать нагрузку на систему, если WhisperX запускается без GPU.
Например:
 - Использовать параметр `--compute_type int8` или «int8» в Python — это снижает требования к памяти.
 - Уменьшить размер модели: вместо `large-v2` попробовать `base` или `medium`.
 - Снижать размер батча: `--batch_size 2` или даже 1, если возникают ошибки CUDA.
- Для CPU-систем — установить `ffmpeg` и запускать WhisperX с `--compute_type int8 --device cpu`.
- Для диаризации — включить токен Hugging Face (`--hf_token`) и принять пользовательское соглашение для моделей Segmentation и Speaker-Diarization.

[aigenom.ru](#) [github.com](#)

Выбор модели

- Использовать модели, совместимые с WhisperX для диаризации.
Например, для `en`, `fr`, `de`, `es`, `it` модели загружаются из `torchaudio` автоматически, для других языков — необходимо указать `--align_model` вручную, выбрав подходящую модель с Hugging Face. Список поддерживаемых языков можно найти в переменной `DEFAULT_ALIGN_MODELS_HF` в `alignment.py`. [aigenom.ru](#) [github.com](#)

Но когда я всё починил и заставил систему работать, результат превзошёл мои ожидания

таким углом, о котором я сам бы никогда не задумался.

Именно в этот момент мой скепсис в отношении «умных ИИ-диктофонов», которые я критиковал в первой статье, сильно пошатнулся. Скорее всего их сила не в тотальной записи, а в возможности превращать хаос в структурированные данные, готовые для анализа.

В этой статье я хочу поделиться своим опытом прохождения этого квеста, показать, как обойти все скрытые сложности, и дать вам готовые инструкции, чтобы вы тоже могли превращать свои записи в осмыслиенные диалоги.

Весь [код выложен на Гитхаб](#).

Platform Solutions Resources Open Source Enterprise Pricing Search or jump to... Sign in Sign up Notifications Fork 4 Star 10

Code Issues Pull requests Actions Projects Security Insights

main 1 Branch 0 Tags Go to file Code About

empenoso Update README.md e24c33f · 2 hours ago 10 Commits

LICENSE Initial commit 2 weeks ago

README.md Update README.md 2 hours ago

whisper_transcribe.py new 2 weeks ago

whisper_transcribe_setup.sh upd last week

whisperx_diarization.py Обновление. Распознаёт отлично yesterday

whisperx_diarization_setup.sh Обновление. Распознаёт отлично yesterday

README MIT license

Локальная система распознавания и диаризации речи (Whisper & WhisperX)

Этот репозиторий содержит набор скриптов для создания полностью автономной системы расшифровки аудиозаписей на вашем компьютере. Решение позволяет не только превращать речь в текст, но и автоматически разделять его по говорящим (диаризация), что идеально подходит для анализа лекций, интервью и совещаний.

В основе лежат мощные open-source модели OpenAI Whisper и WhisperX, что гарантирует высокую точность и полную приватность ваших данных.

Ключевые возможности

- Два режима работы:

whisper whisperx

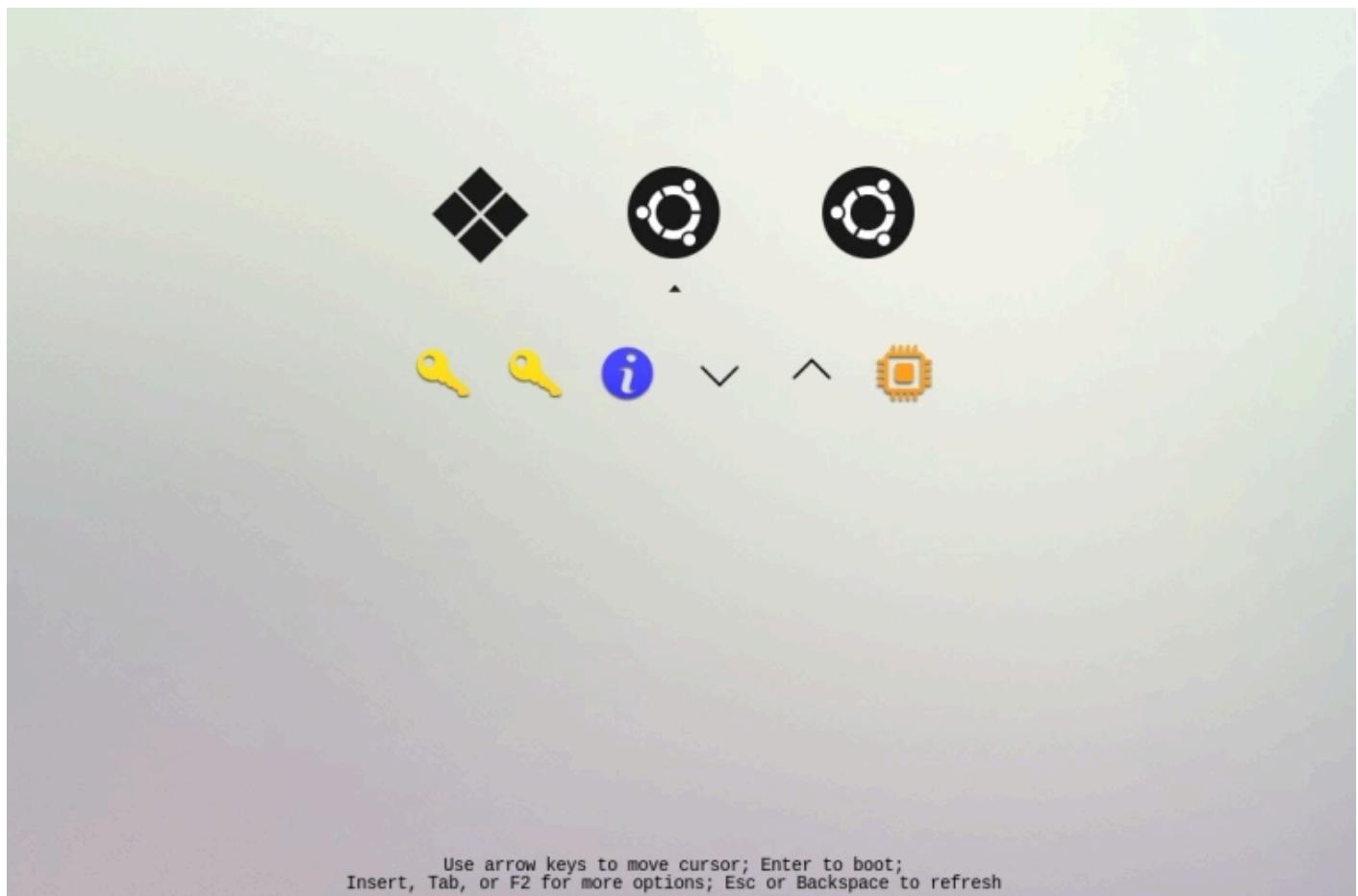
Readme MIT license Activity 10 stars 1 watching 4 forks Report repository

No releases published

No packages published

Python 61.8% Shell 38.2%

на домашнем компьютере стоит Ubuntu в режиме двойной загрузки — и многим непонятно, почему я не сделал всё под Windows. Ответ прост: для задач с нейросетями Linux даёт меньше неожиданностей и больше контроля. Драйверы, контейнеры, права доступа — под Linux их проще исследовать и чинить, особенно когда начинаешь ковырять CUDA и системные зависимости.



Ещё меня критиковали за RTX 5060 Ti 16GB — мол, не у всех такие видеокарты. Согласен, это не смартфон в кармане. Но для работы с большими моделями и диализацией нужна мощь GPU: я использую её как инструмент. К тому же подходы, которые я описываю, работают и на более скромных конфигурациях — просто медленнее.

А теперь начнём с самого начала — что такое Docker простыми словами? Представьте, что вместо того, чтобы настраивать компьютер под каждую программу, вы берёте готовую «коробку» и в ней уже есть всё: нужные версии Python, библиотеки, утилиты. Эта «коробка» запускается одинаково на любой машине — как виртуальная мини-кухня.

То есть мой план действий был такой:

Так что могло пойти не так?

Первое столкновение с реальностью

Уже на первом шаге начались сюрпризы:

Секретный токен, который не дошёл до адресата

Чтобы запустить диаризацию, WhisperX использует модели от ruannote, а они требуют авторизации через токен Hugging Face. Я передал его как переменную окружения Docker (-e HF_TOKEN=...), будучи уверенным, что этого достаточно. Но утилита внутри контейнера ожидала его совсем в другом виде — аргументом командной строки (--hf_token). В итоге модель упорно отказывалась работать, и я долго не понимал, где ошибка.

Война за права доступа

Следующая засада — `PermissionError` при попытке записи в системные папки /.cache. Контейнер как гость в доме: ему разрешили пользоваться кухонным столом, а он пошёл сверлить стены в гостиной. Разумеется, система его остановила. Решение оказалось простым — создать отдельную «полку» для кеша (~/.whisperx) и явно указать путь.

Загадочное зависание

Запускаешь скрипт — и тишина. Ни ошибок, ни логов, будто процесс замёрз. На деле работа шла, просто механизм вывода в контейнере «затыкался». Решение — добавить индикатор прогресса.

Так что Docker — не магия, а всего лишь ещё один инструмент, который тоже нужно приручить.

Решение: два скрипта

Я написал две утилиты — один раз подготовить систему, второй — управлять обработкой. Это простая, надёжная пара: установщик устраняет системные «подводные камни», оркестратор — закрывает все проблемы запуска (HF-token, кэш, права, прогресс).

```
146 log "Конфигурирование Docker для работы с NVIDIA GPU..."
147 sudo nvidia-ctk runtime configure --runtime=docker
148
149 log "Перезапуск Docker daemon для применения конфигурации..."
150 sudo systemctl restart docker
151 sleep 3 # Даем демону время на перезапуск
152 success "Docker настроен для работы с NVIDIA GPU."
153
154 }
155 test_docker_gpu() {
156     log "Тестируем Docker с поддержкой GPU..."
157     if ! sudo docker run --rm hello-world > /dev/null 2>&1; then
158         error "Базовый Docker не работает. Проверьте 'systemctl status docker'"
159         exit 1
160     fi
161     success "Базовый тест Docker пройден."
162
163     log "Проверка доступа к GPU из контейнера..."
164     local cuda_image="nvidia/cuda:12.4.1-base-ubuntu22.04" # Используем актуальный образ
165     log "Используем тестовый образ: $cuda_image"
166
167     if ! sudo docker pull "$cuda_image" > /dev/null; then
168         warning "Не удалось загрузить тестовый образ $cuda_image. Пропускаем тест GPU."
169         return 1
170     fi
171
172     # Пытаемся выполнить nvidia-smi внутри контейнера
173     local gpu_name_in_container
174     gpu_name_in_container=$(sudo docker run --rm --gpus all "$cuda_image" nvidia-smi --query-gpu=name --format=csv,noheader)
175
176     if [[ -n "$gpu_name_in_container" ]]; then
177         success "GPU успешно обнаружен в Docker контейнере: $gpu_name_in_container"
178         return 0 # Успех
179     else
180         error "Не удалось получить доступ к GPU из Docker контейнера."
181         warning "WhisperX будет работать на CPU (значительно медленнее)."
182         log "Возможные причины:"
183         log "- Конфликт версий драйвера, toolkit или docker."
184         log "- Необходимо перезагрузить систему: 'sudo reboot'"
185         return 1 # Неудача
186     fi
187 }
188
189 pull_whisperx_image() {
190     log "Загрузка Docker образа WhisperX..."
191     local whisperx_image="ghcr.io/jim60105/whisperx:latest"
192
193     if sudo docker pull "$whisperx_image"; then
194         success "Образ $whisperx_image загружен успешно."
195         local image_size_bytes
```

Шаг 1. Фундамент: `whisperx_diarization_setup.sh`

Назначение: однократно подготовить Ubuntu – поставить Docker, NVIDIA toolkit, скачать образ WhisperX, создать рабочие папки и общий кэш `~/whisperx`.

Что делает:

- проверяет дистрибутив и наличие GPU (nvidia-smi);
 - устанавливает Docker и добавляет пользователя в группу docker;
 - ставит NVIDIA Container Toolkit и настраивает runtime;
 - подтягивает образ ghcr.io/jim60105/whisperx:latest;
 - создаёт ./audio, ./results и ~/.whisperx, выставляет права и генерирует config.env.

```
cat > ./config.env << 'EOF'  
HF_TOKEN=your_token_here  
WHISPER_MODEL=large-v3  
DEVICE=cuda  
...  
EOF  
# загрузка образа  
sudo docker pull ghcr.io/jim60105/whisperx:latest</code>
```

Шаг 2. Пульт управления: whisperx_diarization.py

Роль: оркестратор – перебирает файлы, формирует корректную команду docker run и решает описанные проблемы. Как он их решает:

- HF_TOKEN передаётся и как -e HF_TOKEN=..., и в аргументах --hf_token при запуске whisperx;
- глобальная папка кеша ~/whisperx монтируется в контейнер и назначается HOME=/models, XDG_CACHE_HOME=/models/.cache – проблем с PermissionError нет;
- прогресс-бар: чтение stderr контейнера и человеко-понятные статусы (VAD → транскрибация → выравнивание → диаризация);
- проверка готовности: --check тестирует Docker, образ и права записи.

Пример:

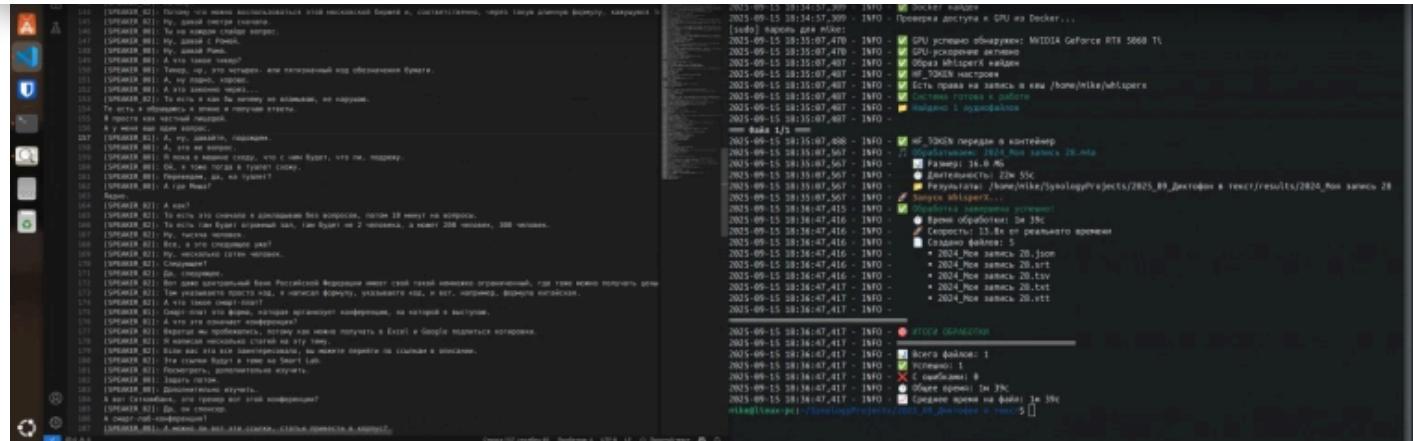
```
<code class="cpp"># проверка системы  
python3 whisperx_diarization.py --check  
# обработать всю папку  
python3 whisperx_diarization.py</code>
```

Подробная инструкция и актуальные скрипты – в репозитории:

👉 github.com/empenoso/offline-audio-transcriber

Результаты

Когда все технические баталии были позади, я наконец смог оценить, стоила ли игра свеч. Результат был отличный.



В первой статье обычный Whisper выдавал сплошное текстовое полотно. Информативно, но безжизненно. Вы не знали, где заканчивается мысль одного человека и начинается реплика другого.

Было (обычный Whisper):

"

... да, я согласен с этим подходом но нужно учесть риски которые мы не обсудили например финансовую сторону вопроса и как это повлияет на сроки я думаю нам стоит вернуться к этому на следующей неделе...

"

Стало (WhisperX с диаризацией):

"

[00:01:15.520 --> 00:01:19.880] SPEAKER_01: Да, я согласен с этим подходом, но нужно учесть риски, которые мы не обсудили.

[00:01:20.100 --> 00:01:22.740] SPEAKER_02: Например, финансовую сторону вопроса и как это повлияет на сроки?

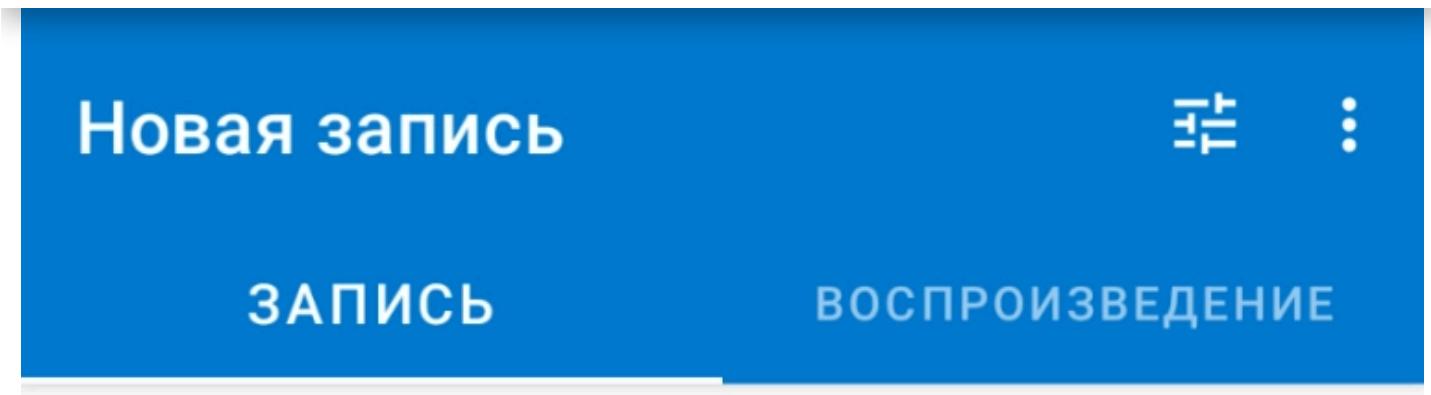
[00:01:23.020 --> 00:01:25.900] SPEAKER_01: Именно. Я думаю, нам стоит вернуться к этому на следующей неделе.

"

WhisperX с диаризацией превращает этот монолит в сценарий пьесы. Каждый спикер получает свой идентификатор, а его реплики – точные временные метки. Разница колоссальная. Теперь это не просто расшифровка, а полноценный протокол.

этот диалог».

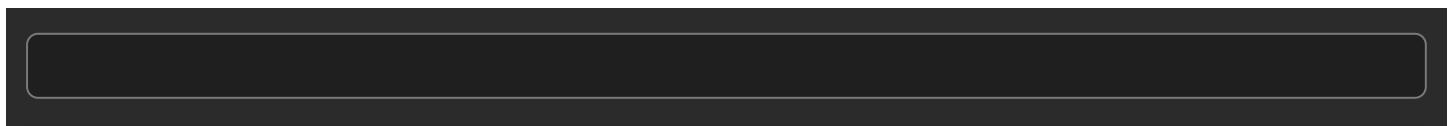
Именно из-за структуры Gemini смогла отследить, кто инициировал темы, кто чаще соглашался или перебивал, как менялась тональность и динамика беседы. В итоге я получил анализ скрытых паттернов в общении, о которых сам никогда бы не задумался. Это был взгляд на ситуацию с абсолютно неожиданной стороны, который помог мне лучше понять и себя, и собеседника.



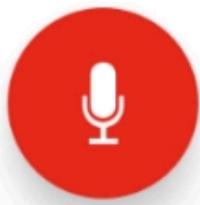
Загрузка рекламы...

Моя запись 8.m4a

Готова к записи



~929ч свободного места осталось



Даже бесплатное приложение в телефоне может служить источником

Я понял, что их главная ценность «ИИ-диктофонов» — не в способности записывать каждый ваш шаг, а в умении превращать хаос человеческого общения в структурированные, машиночитаемые данные. Это открывает возможности: от создания кратких сводок по итогам встреч до глубокого анализа коммуникаций, который раньше был невозможен.

Заключение

В итоге путь от «просто используй Docker» к рабочей связке WhisperX показал очевидную вещь: контейнеры — удобный инструмент, но не магия.

Подготовка системы и правильная оркестровка запуска — это то, что превращает хаос в рабочий процесс. Если вы готовы потерпеть небольшие сложности ради удобства в дальнейшем — результат оправдает усилия: структурированные протоколы и возможность глубокого анализа бесед.

Автор: Михаил Шардин

OpenAI

ChatGPT

Linux

2.3K



5

6

11



Михаил Шардин

Пермь

398 4 228

с 23 января 2019

+ Подписаться

6 КОММЕНТАРИЕВ

[Сначала старые ▼](#)

Trader_Khv

23 сентября 2025, 05:06

⋮

Прочитал только начало... У меня Galaxy 23Ultra делает транскрибацию по голосам без всяких танцев с бубном, запись звонка автоматически. Только, конечно ии распознает не идеально, есть косяки, но 80% вполне понятно

Показать 1 ответ

+1



Gorizont_*^*_

23 сентября 2025, 08:47

⋮

какой LLM при написании кода пользуетесь?)

Показать 3 ответа

Напишите комментарий...



ОТПРАВИТЬ



Формат QSH в OsEngine: поддержка и особенности работы

Недавно OsEngine начал поддержку бинарного формата хранения и трансляции данных по стаканам. Это было нужно, чтобы:...

 OsEngine

18:01

Промышленная автоматизация – один из ключевых трендов 2026 в ИТ



Промышленная автоматизация – один из ключевых трендов 2026 в ИТ #SOFL_тренды

Сегодня промышленность все чаще смотрит на ИТ как на инструмент для наращивания мощностей. Для российской...

 Softline

17:16

ОБЛИГАЦИИ
ПСБ ФИНАНС
20 ЯНВАРЯ



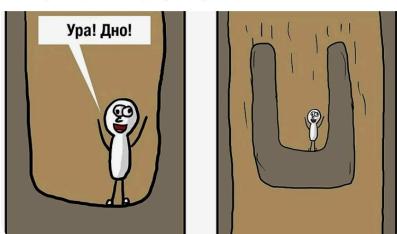
⚡ Объявляем условия нового размещения

20 января финтех-сервис ПСБ Финанс (бренд CarMoney) начнет размещение нового выпуска облигаций с плавающим купоном...

 CarMoney

17:52

кабре идет ко дну - хуже не было никогда



Хэдхантер. Ситуация на рынке труда в декабре идет ко дну - хуже не было никогда

Вышла статистика рынка труда за декабрь 2025 года, которую Хэдхантер публикует ежемесячно, что же там интересного:...

 Mozgovik

13.01.2026

Установите приложение Смартлаба:



RuStore



AppGallery



App Store

