



Горячее

Лучшее

Свежее

...



empenoso 3 часа назад



Программирование на python

## Умный диктофон без облака - или как я приручал Docker

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. Тогда статья нашла своего читателя, собрав 16K просмотров и 120 закладок.

**Plaud note pin**  
Умный диктофон Plaud Note Pin (Cosmic gray) с чатом GPT - диктофон с искусственным интеллектом...  
★ 4.9 ● 158 отзывов  
18126 Р  
18 сентября

**ПРОФЕССИОНАЛЬНЫЙ ДИКТОФОН KR-666**  
СХЕМАТЕХ  
7990 Р  
20 сентября

**Диктофон профессиональный**  
100% время записи  
32 Гб объем памяти  
3179 Р 9990 Р -68%  
61 шт осталось  
СХЕМАТЕХ  
Диктофон профессиональный V65 / MP3-плеер / 32GB  
★ 4.9 ● 90 отзывов  
20 сентября

**ДИКТОФОН**  
с искусственным интеллектом  
расшифровка записей  
преобразует аудио в текст  
19301 Р 59990 Р -67%  
70 шт 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 собирал 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 текста, а готовый протокол встречи, где понятно, кто и что сказал. Казалось, с Docker это будет легко.

Но я заблуждался. Путь «в одну команду» оказался полон сюрпризов - всё сыпалось одно за другим: то скрипт не видел мои файлы, то не мог получить к ним доступ, то



Используйте аккаунт Яндекса  
для входа на сервис

Безопасный вход без дополнительной  
регистрации на сайте

Войти с Яндекс ID

### Войти

Войти

Создать аккаунт

[Забыли пароль?](#)

или продолжите с



Войти с Яндекс ID



Войти через VK ID



Промокоды



Работа



Курсы



Реклама



Игры



Пополнение Steam



просто зависал без объяснения причин. Внутри этой «волшебной упаковки» царил хаос, и мне приходилось разбираться, почему она не хочет работать.

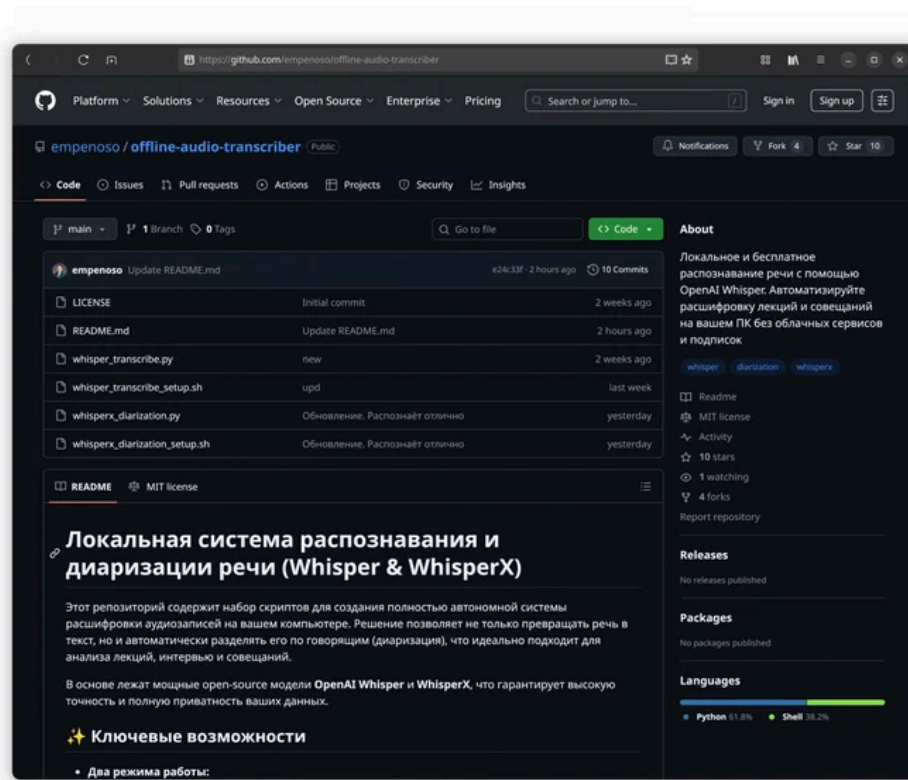
Но когда я всё починил и заставил систему работать, результат превзошёл мои ожидания. Новейшая модель large-v3 в связке с диаризацией выдала не просто текст, а структурированный диалог.

Это был настолько лучший результат, что я смог передать его большой языковой модели (LLM) и получить глубокий анализ одной очень важной для меня личной ситуации - под 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 осмысленные диалоги.

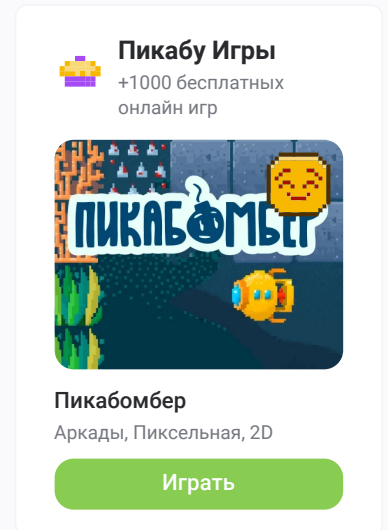
Весь код выложен на Гитхаб.



Код на Гитхабе

## Docker как панацея и почему Linux

В комментариях меня критиковали за то, что я опять написал статью про Linux. Да, у меня 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 диаризацией 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:

1. Установить Docker.
2. Скачать готовый образ с WhisperX.
3. Запустить одну команду и получить готовый протокол встречи.

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 ryannote, 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.



#### Топ прошлой недели

- ExtremeGrower  
6 постов
- ROMBOR  
9 постов
- Lio2142  
1 пост

[Посмотреть весь топ](#)

#### Лучшие посты недели

Рассылка Пикабу: отправляем самые рейтинговые материалы за 7 дней 🔥

Укажи [Подписаться](#)

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



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).

### Шаг 1. Фундамент: `whisperx_diarization_setup.sh`

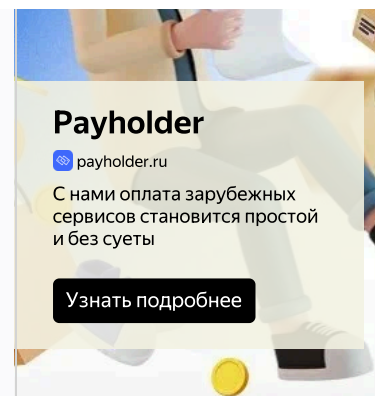
Назначение: однократно подготовить Ubuntu - поставить Docker, NVIDIA toolkit, скачать образ WhisperX, создать рабочие папки и общий кэш `~/whisperx`.

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`.

Пример:

```
# создаём директорию и конфиг
mkdir -p ./audio ./results "$HOME/whisperx"
chmod -R 777 ./audio ./results "$HOME/whisperx"
cat > ./config.env <<'EOF'
```



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

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



РЕКЛАМА



```
HF_TOKEN=your_token_here
WHISPER_MODEL=large-v3
DEVICE=cuda
...
EOF
# загрузка образа
sudo docker pull ghcr.io/jim60105/whisperx:latest
```

## 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:

```
# проверка системы
python3 whisperx_diarization.py --check
# обработать всю папку
python3 whisperx_diarization.py
```

Подробная инструкция и актуальные скрипты - в репозитории:

👉 <https://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

**M W**

MTC WEI

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?

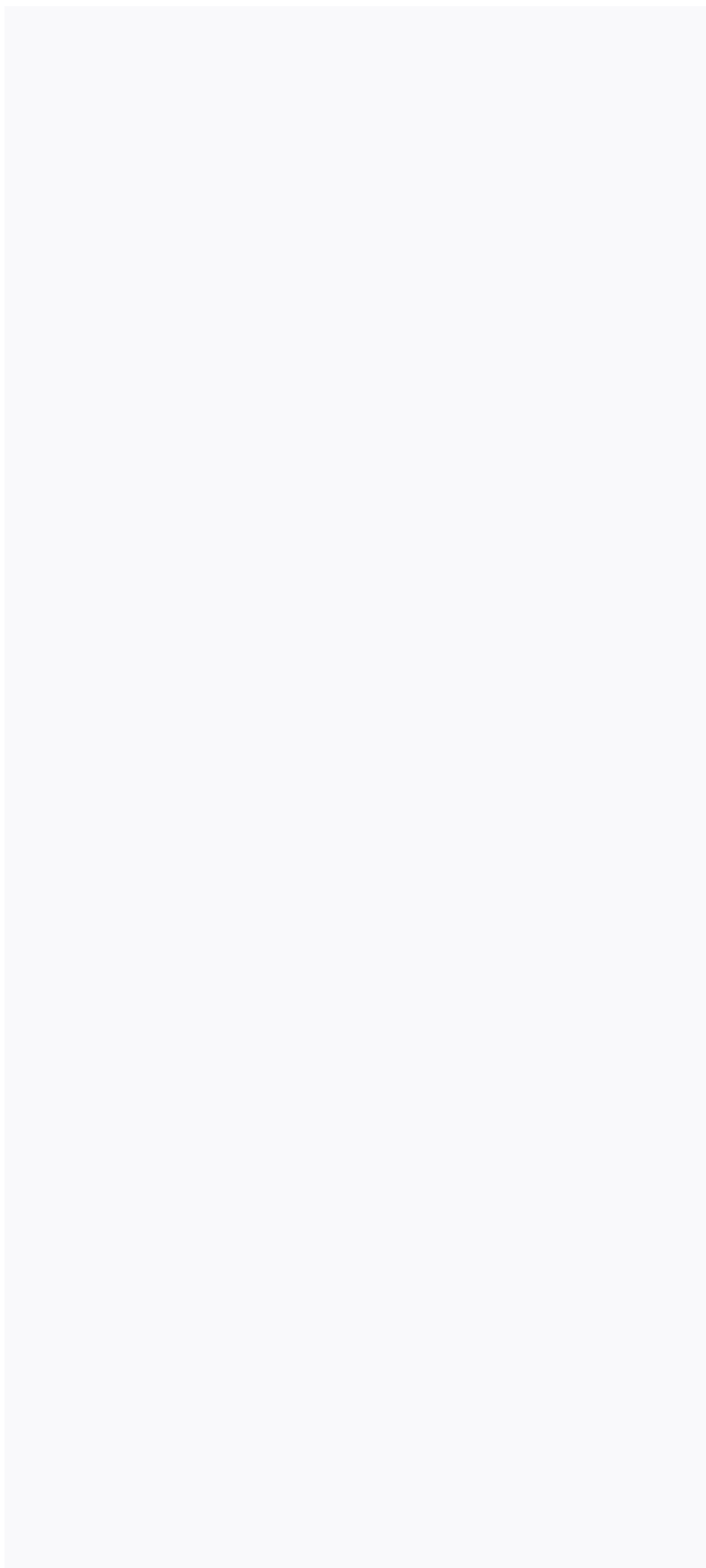
[00:01:23.020 -> 00:01:25.900] SPEAKER\_01: Именно. Я думаю, нам стоит вернуться к этому на следующей неделе.

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.

#### Мой личный кейс

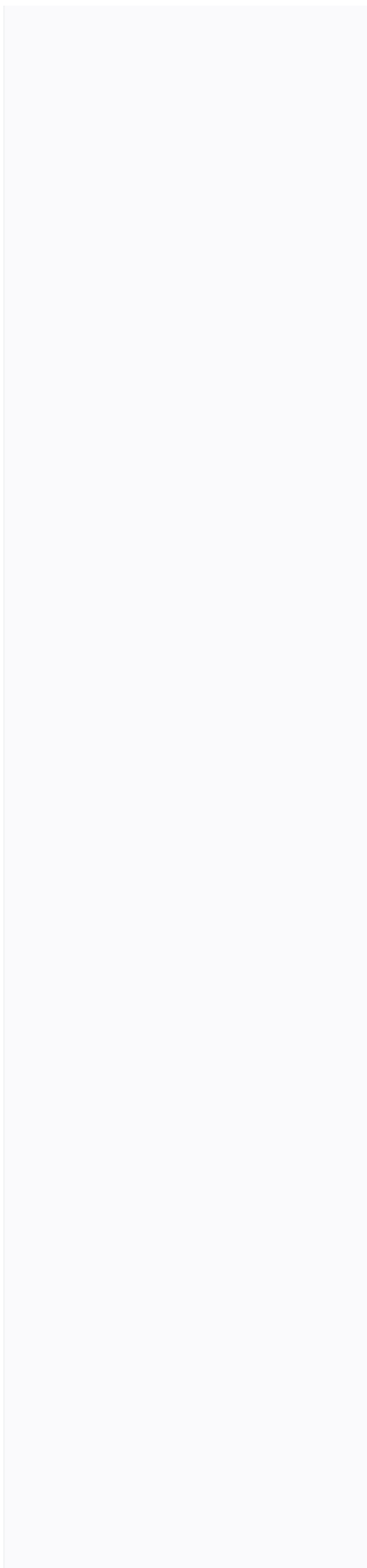
Но настоящая магия началась, когда я решил пойти дальше. Я взял расшифровку одного личного разговора, сохранённую в таком структурированном виде, и загрузил её в нейросеть бесплатную нейросеть от гугла Gemini 2.5 Pro с простым запросом: «Действуй как аналитик. Проанализируй этот диалог».

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.



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.

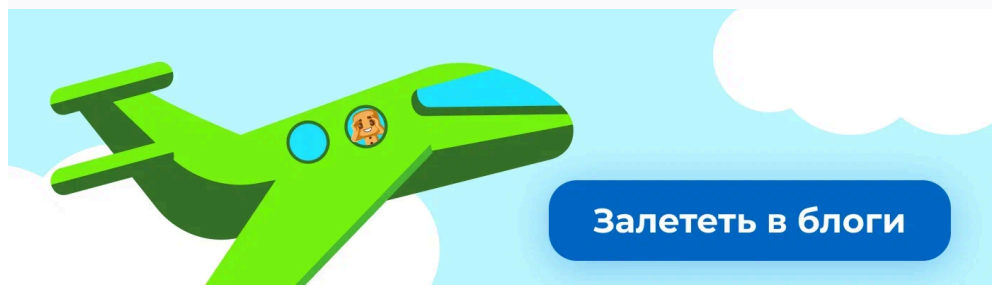
Подготовка системы и правильная оркестровка запуска - это то, что превращает хаос в рабочий процесс. Если вы готовы потерпеть небольшие сложности ради удобства в дальнейшем - результат оправдает усилия: структурированные протоколы и возможность глубокого анализа бесед.

**Автор:** Михаил Шардин

 [Моя онлайн-визитка](#)

 [Telegram «Умный Дом Инвестора»](#)

23 сентября 2025



**Программирование на python**

893 поста • 11.9K подписчиков

[Добавить пост](#)

[Подписаться](#)



### Правила сообщества

Публиковать могут пользователи с любым рейтингом. Однако!

Приветствуется:...

[Подробнее](#) 

[Все комментарии](#) [Автора](#)



Раскрыть 3 комментария

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

● — ■ —  
—  
—  
—  
—  
—  
—

● — ■ —  
—  
—  
—  
—  
—  
—

● — ■ —  
—  
—  
—  
—  
—  
—

