

РЕКЛАМА

Курсы на Хабр Карьере

Хабр

Курсы — инвестиция в себя



empenoso

2 сен 2025 в 05:23

Как локально и бесплатно распознать текст лекции или совещания и делать это регулярно

Простой 6 мин 14К

Open source*, Настройка Linux*, Умный дом, Python*

Кейс

На онлайн-площадках и ИИ-диктофоны и обычные попеременно продаются

В новостях всё чаще [говорят об «ИИ-диктофонах»](#) — гаджетах, которые записывают **каждый** ваш разговор в течение дня, отправляют аудио в облако, превращают его в текст и даже готовят краткую сводку по итогам. Звучит футуристично, но такие решения стоят дорого, требуют постоянной подписки и вызывают вопросы о приватности.

Лично мне идея тотальной записи кажется избыточной. Зато куда практичнее другая задача: получить точную текстовую расшифровку лекции, доклада или публичного выступления. Чтобы потом не переслушивать часы аудио, а быстро найти нужную цитату или мысль простым поиском по тексту.

В этой статье я покажу, как построить такую систему без платных подписок и полностью под вашим контролем. Всё, что нужно — обычный диктофон за 1–3 тыс. рублей или даже просто приложение на телефоне — тогда затраты вообще равны нулю, и набор бесплатных, открытых программ, которые работают на вашем компьютере. Я купил диктофон для теста и поделюсь результатами.



Мой купленный за 2 т.р. диктофон с возможностью подключения внешнего микрофона на фоне коробки с ESP32

Сердцем решения станет [OpenAI Whisper](#) — мощная технология распознавания речи от создателей ChatGPT. Главное её преимущество — она может работать полностью автономно на вашем ПК, не отправляя никуда ваши данные. К тому же Whisper распространяется как open-source: исходный код и модели доступны бесплатно — вы можете скачать, использовать и при необходимости даже модифицировать.

Мои скрипты [выложены на GitHub](#).

Теоретическая часть: что, почему и как?

За последние пару лет появилось немало open-source решений для распознавания речи, но именно Whisper стал фактическим стандартом. Его модели обучены на колоссальном массиве данных, что обеспечивает высокую точность распознавания. По сравнению с другими бесплатными движками, Whisper даёт результат ближе всего к коммерческим сервисам вроде Google Speech-to-Text и при этом работает автономно. Важный плюс — мультиязычность. Русский язык поддерживается «из коробки».

Модели Whisper бывают разных размеров: от tiny до large. На данный момент наиболее актуальной и точной является large-v3. Главный принцип здесь — компромисс между скоростью, точностью и требуемыми ресурсами (в первую очередь, видеопамятью). У меня [видеокарта NVIDIA GeForce RTX 5060 Ti 16 ГБ](#), поэтому на тестах использую large модель, она требует ~10 ГБ VRAM, но можно начать и со small модели — для неё достаточно ~2 ГБ VRAM.

Не стоит забывать и о приватности: все данные остаются у вас на компьютере. Никаких облачных серверов, никаких подписок. Что понадобится для запуска?

Железо: компьютер с Linux (я использую Ubuntu, но [у меня стоит двойная загрузка Windows & Linux через rEFInd Boot Manager](#)). Рекомендуются видеокарта NVIDIA — GPU многократно ускоряет

работу, хотя на CPU тоже всё запустится, только медленнее. В качестве источника звука я тестировал обычный диктофон за пару тысяч рублей.

Диктофон за 1–3 тыс. рублей. Много их

Софт:

- *Python* — язык, на котором работает весь стек.
- *FFmpeg* — универсальный конвертер аудио/видео.
- *PyTorch* — фреймворк, на котором обучены модели.
- *NVIDIA Drivers и CUDA* — для связи с видеокартой.

Практическая часть: пошаговая инструкция

Теперь перейдём от теории к практике и соберём рабочую систему распознавания. Я разбил процесс на несколько шагов — так будет проще повторить.

Анализ лиц с домофона: как я победил несовместимости и собрал dlib+CUDA на Ubuntu ...

Каждый день мимо двери моего подъезда проходят десятки людей. Иногда это знакомые соседи, но чаще — ...

habr.com



Шаг 1. Подготовка окружения

Когда-то я собирал dlib с поддержкой CUDA для анализа лиц с камеры в подъезде. Тогда я прошёл через несовместимости, конфликты версий и ручную сборку библиотек. Поэтому к установке Whisper я уже был подготовлен.


Чтобы избавить вас от всего этого «удовольствия», я написал универсальный [bash-скрипт setup_whisper.sh](#). Он берёт на себя всю грязную работу по настройке окружения на Ubuntu 24:

- обновляет систему и ставит базовые пакеты, включая Python и FFmpeg;

- проверяет драйверы NVIDIA и при необходимости устанавливает их;
- подтягивает CUDA Toolkit;
- создаёт виртуальное окружение Python и внутри него ставит PyTorch (учитывая модель видеокарты);
- загружает сам Whisper и полезные библиотеки;
- запускает тест, проверяющий, что GPU действительно работает.

Запуск прост:

```
chmod +x setup_whisper.sh
./setup_whisper.sh
```

Объяснить код с  SourceCraft

Запуск ./setup_whisper.sh

Полный [код setup_whisper.sh](#) на Гитхабе.

Шаг 2. Запись и подготовка аудио

Чем лучше исходная запись, тем меньше ошибок. Записывайте ближе к источнику звука, избегайте шумных помещений и треска. Whisper работает с самыми популярными форматами: mp3, wav, m4a, так что конвертировать вручную не придётся.

Шаг 3. Массовая расшифровка всех подряд записей

Здесь в игру вступает мой [второй скрипт](#) — `whisper_transcribe.py`. Он:


- автоматически находит все аудиофайлы в папке;
- использует GPU (если доступен), ускоряя работу в десятки раз;
- сохраняет результат в нескольких форматах:
 - `.txt` для текста,
 - `.srt` с таймкодами (можно открыть как субтитры),
 - `all_transcripts.txt` — общий файл со всеми расшифровками.

Пример использования:

```
# Активируем окружение
source .venv/bin/activate

# Запуск по умолчанию (ищет аудио в текущей папке)
python3 whisper_transcribe.py

# Указываем папку с файлами, модель и папку для результатов
python3 whisper_transcribe.py ./audio large ./results
```

Объяснить код с  SourceCraft

Полный код [whisper_transcribe.py](#) на Гитхабе.


```
python3 whisper_transcribe.py ./audio large ./results
```

Шаг 4. Анализ результатов

После обработки вы получите полный набор файлов. Например:

- `some_lecture.txt` — текст лекции;
- `some_lecture.srt` — субтитры вида:

```
12
00:04:22,500 --> 00:04:26,200
Здесь спикер рассказывает о ключевой идее...
```

Объяснить код с  SourceCraft

- `all_transcripts.txt` — всё сразу в одном документе.

Я проверил систему на часовом файле. Модель **large** на моей RTX 5060 Ti справилась за ~8 минут.

Разделение по спикерам (диаризация) - почему это сложно?

А если записывать не лекцию, а совещание? На записи говорят пять человек, и вам нужно понять, кто именно что сказал. Обычный Whisper выдаёт сплошной текст без указания человека. Здесь на помощь приходит диаризация — технология, которая анализирует голосовые характеристики и помечает фрагменты как «Спикер 1», «Спикер 2» и так далее.

Для этого существует WhisperX — расширенная версия Whisper с поддержкой диаризации. **Однако при попытке установки я опять столкнулся с классической проблемой ML-экосистемы: конфликтом зависимостей. WhisperX требует определённые версии torchaudio**, которые несовместимы с новыми драйверами NVIDIA для RTX 5060 Ti.

Решение мне подсказали: Docker-контейнеры NVIDIA. По сути, это готовые «коробки» с предустановленным софтом для машинного обучения — разработчики уже решили все проблемы совместимости за вас. NVIDIA поддерживает целую экосистему таких контейнеров через NGC (NVIDIA GPU Cloud), а сообщество создает специализированные образы под конкретные задачи. Вместо многочасовой борьбы с зависимостями достаточно одной команды `docker pull`, и вы получаете полностью рабочую среду с предустановленным WhisperX, настроенным PyTorch и всеми библиотеками. В данном случае контейнер ghcr.io/jim60105/whisperx включает диаризацию из коробки и отлично работает с современными GPU.

Диаризация откроет новые возможности: автоматическую генерацию протоколов встреч с указанием авторства реплик, анализ активности участников дискуссий, создание интерактивных расшифровок с навигацией по спикерам.

Это тема для отдельной статьи, которую планирую выпустить после тестирования Docker-решения на реальных многоголосых записях.

⚠ Уже вышло продолжение: <https://habr.com/p/948894/>

Заключение

Мы собрали систему, которая позволяет бесплатно и полностью автономно расшифровывать лекции, выступления, а в перспективе и совещания. В основе — OpenAI Whisper, а все настройки и запуск упрощают мои open source скрипты. Достаточно один раз подготовить окружение — и дальше вы сможете регулярно получать точные транскрипты без подписок и риска приватности.

Следующий шаг — диаризация. Это позволит автоматически разделять текст по спикерам и превращать расшифровку совещания в полноценный протокол с указанием авторства.

Автор: Михаил Шардин

 [Моя онлайн-визитка](#)

 [Telegram «Умный Дом Инвестора»](#)

2 сентября 2025

Теги: диктофон, whisper, Whisperx, openai, rtx 5060, ngc, rtx, cuda, расшифровка аудио

Хабы: Open source, Настройка Linux, Умный дом, Python

Редакторский дайджест

Присылаем лучшие статьи раз в месяц

Подписаться

Оставляя почту, я принимаю [Политику конфиденциальности](#) и даю согласие на получение рассылок



256

355.6

Карма

Общий рейтинг

Михаил Шардин @empenoso

Автоматизация / Data & ML / Финансы / Smart Home

Подписаться



[Сайт](#) [Сайт](#) [GitHub](#)

 Комментарии 12

Публикации

ЛУЧШИЕ ЗА СУТКИ

ПОХОЖИЕ

**DrArgentum**

7 часов назад

Ненормальные непотребства, трюки, хаки и алгоритмы на C

 Простой 10 мин 3.7K

Обзор

 +25 32 9**Sivchenko_translate**

6 часов назад

std::move ничего никуда не двигает: подробный рассказ о категориях значений в C++

 35 мин 3.4K

Перевод

 +21 34 2**TrexSelectel**

6 часов назад

Полезные ресурсы для тестировщиков: подборка от специалистов Selectel

 3 мин 2.7K

Мнение

 +21 7 0**shiru8bit**

6 часов назад

Игра во время загрузки игры

 Простой 15 мин 3K

Ретроспектива

 +19 3 2

CreatorLAB

22 часа назад

Конфигуратор микроконтроллеров STM8S103/105

 10 мин 10K +18 39 8

dronnix

8 часов назад

Black-White Array: новая структура данных с $O(\log N)$ аллокаций

 Средний 8 мин 5.1K

Обзор

 +17 39 2

beget_com

8 часов назад

Автомобили-конструкторы, кафе с удалёнными официантами и отстреливающиеся батареи: 15 проектов промдизайна 2025

 6 мин 5.2K

Кейс

 +16 2 1

Grishandin

8 часов назад

Закономерности в данных вместо догадок: как мы помогаем студентам дойти до конца курса

 Средний 8 мин 3.9K

Кейс

 +16 5 0

sergbe

8 часов назад

Рецензия на книгу «Принципы модернизации программных архитектур»

 Простой 9 мин 4.8K

Мнение

 +14 7 0

stasvetokhin

9 часов назад

Как выбрать идею для инди-игры и не потратить годы впустую

 Простой 8 мин 5K

Кейс

+13

10

2

Как начать инвестировать и на что обращать внимание — ответы в статье

[Промо](#)[Показать еще](#)

КУРСЫ

 1C-программист

По мере набора группы

 Python-разработчик

По мере набора группы

 Frontend-разработчик

По мере набора группы

 Профессия Графический дизайнер PRO

По мере набора группы

 HR Бизнес-Партнер

По мере набора группы

[Больше курсов на Хабр Карьере](#)

МИНУТОЧКУ ВНИМАНИЯ



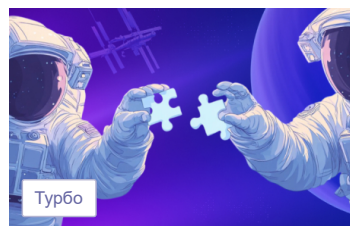
Турбо

От безумных стартапов
до стажировки на «Импульсе Т1»



Событие

Онлайн и офлайн: зимние IT-
ивенты в Календаре



Турбо

Open source как точка роста: какие
проекты получают гранты

БЛИЖАЙШИЕ СОБЫТИЯ



25 ноября 2025 – 16 января 2026

Сезон «ИИ в разработке»

Онлайн

Разработка

Другое

[Больше событий в календаре](#)

Хабр



🌐 [Настройка языка](#)

[Техническая поддержка](#)

© 2006–2026, Habr