Общество с ограниченной ответственностью «Войси»

УТВЕРЖДЕНО

01.07.2024 г.

Программный комплекс распознавания и синтеза речи Voicee (Войси)

ОПИСАНИЕ ПРОГРАММЫ

Листов 12

Москва

2024

АННОТАЦИЯ

Настоящий документ содержит описание Программного комплекса распознавания и синтеза речи Voicee (далее – Voicee, Войси, программа, ПО).

В разделе «Общие сведения» приведена информация об обозначении и наименовании программы; программном обеспечении, необходимом для ее функционирования; а также языках программирования, на которых написана программа.

Раздел «Функциональное назначение» содержит сведения о назначении, классах решаемых задач и области применения программы, а также сведения о функциональных ограничениях на применение.

В разделе «Описание логической структуры» описаны алгоритм программы, используемые методы, а также структура программы с описанием функций составных частей и связей между ними.

В разделе «Используемые технические средства» указаны типы электронно-вычислительных машин и устройств, которые используются при работе с программой.

Раздел «Вызов и загрузка» содержит сведения о способах вызова программы с соответствующего носителя данных, а также входные точки в программу.

В разделах «Входные данные» и «Выходные данные» указаны характер и организация входных и выходных данных соответственно, а также формат, описание и способы их кодирования.

Оформление программного документа «Описание программы» произведено по требованиям ЕСПД ГОСТ 19.402-78.

СОДЕРЖАНИЕ

1 Общие сведения

Полное наименование программы: Программный комплекс распознавания и синтеза речи Voicee (Войси).

Сокращенное наименование программы: ПО Voicee.

Для функционирования программы необходимо наличие веб-браузера или приложения для обмена мгновенными сообщениями Telegram, а также выход в сеть интернет. Для воспроизведения видео или аудио файлов может потребоваться приложение плеер, обычно поставляемое вместе с операционной системой (далее – ОС). В случае его отсутствия рекомендуется установить свободно распространяемое программное обеспечение VLC Media Player или его аналог. Для просмотра и транскрибации документов в формате PDF потребуется веб-браузер или программное обеспечение Adobe Acrobat Reader. Для просмотра и редактирования документов в формате DOCX с последующей транскрибацией потребуется программное обеспечение Microsoft Word. Другие особые программы не требуются.

Основной язык, на котором написано ПО Voicee — Python. В разработке веб-интерфейса применяется фреймворк React и язык программирования JavaScript, а также язык разметки HTML и таблицы стилей CSS.

2 Функциональное назначение

Программный комплекс распознавания и синтеза речи Voicee (Войси) предназначен для транскрибации речи в текстовый формат с использованием комбинации современных нейросетей и алгоритмов, которые обеспечивают быстрое и точное преобразование речи в текст, минимизируя количество возможных ошибок.

Решаемые задачи связаны с обработкой и анализом речи на естественном языке в том или ином виде:

- преобразование речь на записи аудио\видео в текстовый формат;

- обратная задача — синтез речи на естественном языке из текста;

- анализ и пост-обработка теста, полученного из речи для выделения ключевых мыслей, перевода на другие языки и других задач;

- определение характеристик спикера на аудио\видео записи для разделения речи разных спикеров и\или узнавания спикеров на разных записях в т.ч. голосовое подтверждение личности;

- комбинация обозначенных выше задач, открывающая новые функциональные возможности, такие как автоматический перевод и озвучивание видеороликов, сокращение длинных аудиозаписей, например, лекций, с сохранением голоса спикера за счёт его клонирования и т.п.

К числу ограничений можно отнести ограниченный набор языков, который поддерживает ПО Voicee. В текущей версии реализовано 36 языков: русский, арабский, каталанский, чешский, датский, немецкий, греческий, английский, испанский, персидский, финский, французский, иврит, хинди, венгерский, индонезийский, итальянский, японский, казахский, корейский, малаялам, нидерландский, нюнорск (новый норвежский язык),

норвежский, польский, португальский, румынский, словенский, телугу, турецкий, татарский, украинский, урду, узбекский, вьетнамский, китайский.

Набор языков постоянно расширяется и добавляется в том числе по требованиям бизнес-заказчиков.

Также ограничена длина обрабатываемых аудио и видео файлов — максимальная продолжительность 10 часов. Обработка более длинных записей возможна в ручном режиме при наличии таких запросов у заказчика т.е. это не ограничение технологи, но ограничение ПО Voicee в сегменте B2C.

Основные сферы применения ПО Voicee представлены в таблице 1.

Таблица 1 – Основные сферы применения Voicee

Сфера применения	Направления использования
Бизнес и корпоративный сектор	Для транскрибации рабочих совещаний, обучающих семинаров, маркетинговых материалов, телефонных переговоров и собеседований и в качестве инструмента поддержки отделов продаж и обслуживания клиентов.
Юридическая сфера	Для документации заседаний, допросов, свидетельских показаний и других юридически важных событий.
Образование	Для создания текстовых материалов из лекций и образовательных материалов, облегчения доступности контента для иностранных студентов через перевод, адаптации и переозвучивания аудио/видео материалов, документации и архивации результатов научных экспериментов, .лекций, интервью и т.д.
Медиа и развлечения	Для генерации субтитров, переозвучивания и доступности контента на разных языках, транскрибации интервью, подкастов.
Частное использование	Для фиксации заметок, дневников, личного архива аудиозаписей, перевода устного нарратива в текстовый фоормат, создания мультиязычного контента.

3 Описание логической структуры

3.1 Алгоритм программы

Общий порядок работы системы:

- пользователи загружают свои файлы посредством интерфейсов через биллинг в воркеры, которые производят непосредственную обработку по одному из выбранных пользователем сценариев (транскрибация, субтитры, перевод и т.д.);
- биллинг пользователей работает до начала обработки, предусмотрена система покупки секунд обработки, которая записывается конкретному пользователю (физическому либо юридическому лицу) на его внутренний аккаунт. Он производит

также учет всех операций и реализует механизмы возврата средств при неудачной обработке файла;
- если на этапе биллинга не возникло проблем, то файл передается на обработку воркеру. Он производит скачивание исходника, делает предварительную обработку (извлечение аудио дорожки, конвертацию под необходимые параметры), удаляет исходный файл, производит языковую обработку для определения исходного языка после чего делает обработку и возвращает источнику готовый результат.

Этап Language Processing, обработка файла в ядре Worker — основной алгоритм, составляющий научное ноу-хау проекта, состоит из следующих этапов:
- определение участков с человеческой речью во входящем аудиофайле;
- объединение участков с речью в фрагменты для параллельной обработки с целью ускорения работы алгоритма;
- распознавание речи в текст нейросетевой моделью в параллельном режиме;
- оценка качества распознавания речи по фрагментам;
- отправка плохо распознанных предложений на повторную обработку более медленным, но более точным способом;
- повторная оценка качества распознавания фрагментов;
- повторное распознавание фрагментов с низкой оценкой альтернативным способом;
- выбор наилучшего варианта распознавания из всех вариантов;
- восстановление знаков препинания;
- разбивка фрагментов текста на предложения;
- пост-обработка предложений (слияние или разбивка на основе набора правил);
- восстановление временных отметок по словам и предложениям из исходного аудио/видеофайла;
- вычисление характеристик голосов на записи;
- кластеризация голосов для выявления отдельных спикеров;
- предсказание пола и возраста спикеров нейросетевой моделью;
- разметка распознанного текста по спикерам;
- разбиение текста на абзацы по смысловым сегментам.

3.2 Используемые методы

Программа использует следующие методы:

- метод выставления счета и его оплаты посредством телеграм бота;
- метод для скачивания файлов;
- метод обработки файла в нейросетевой модели;
- метод пост-обработки результата обработки нейросетевой модели, преобразование к различным форматам (pdf, txt, docx, xlsx, csv);
- метод работы с базой данных;
- метод обработки и распознавания языка;
- метод обработки и компрессии исходного аудио/видео файла;
- группа методов для работы с интерфейсом бота, в том числе методы для работы с различными сценариями обработки.

3.3 Структура программы с описанием функций основных составных частей и связи между ними:

Программный комплекс Voicee состоит из следующих компонентов (или модулей):

- *Интерфейсы:* Telegram-бот и http api, с которыми взаимодействуют пользователи;
- *Воркер:* система из нескольких сервисов для обработки контента;
- *Биллинг:* система учета пользователей и их баланса средств.

На рисунке 1 показана схема взаимодействия компонентов/модулей программного комплекса Voicee внутри ПО и с внешней средой.

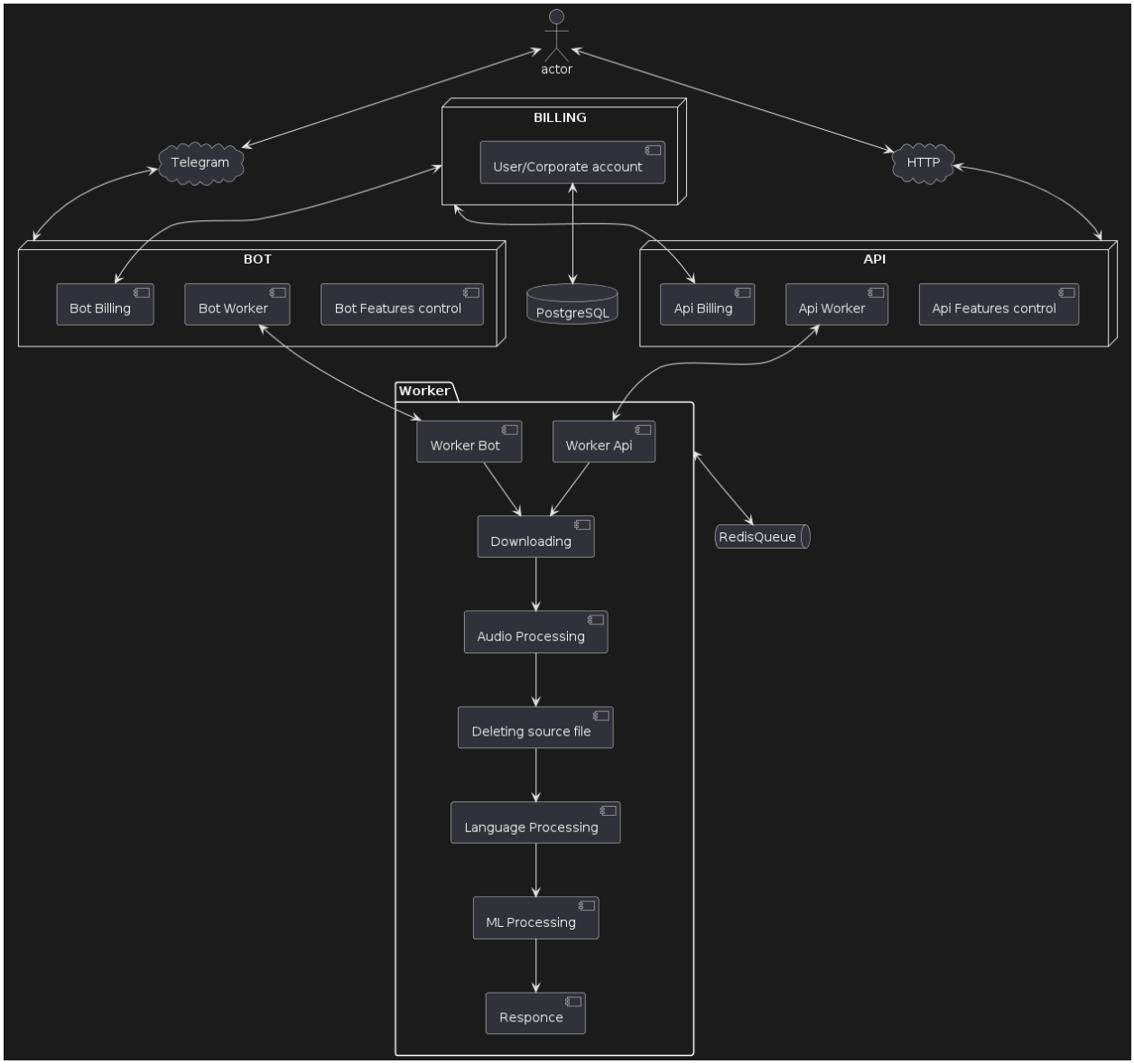

Рисунок 1 – Схема взаимодействия компонентов/модулей программного комплекса Voicee внутри ПО и с внешней средой

4 Используемые технические средства

На клиентской стороне т.е. для пользователей могут использоваться:

- смартфоны на базе ОС Android, iOS, Tizen, Аврора и любых других, которые поддерживают выход в интернет и имеют в составе веб-браузер.

- планшетные компьютеры на базе ОС Android, iOS, Windows с поддержкой выхода в интернет и при наличии веб-браузера.

- настольные ПК и ноутбуки с ОС Windows, macOS, Linux.

- специализированные устройства и одноплатные компьютеры Raspberry Pi или аналоги, позволяющие установить и запустить веб-браузер и имеющие доступ к сети интернет.

На серверной стороне т.е. для клиентов, купивших коробочную версию Voicee для работы в контуре заказчика, используется:

- сервер на базе ОС Linux, поддерживающий контейнеризацию Docker, имеющий не менее 8 ядер процессора уровня Intel Xeon поколения Haswell и выше или аналогичный процессор AMD, с ОЗУ не менее 64 GB и графическим ускорителем с поддержкой технологии CUDA мощностью не ниже, чем Nvidia A40 или аналоги, поколения не ниже Ampere с памятью не менее 40 GB, а также SSD ёмкостью не менее 250 GB.

5 Вызов и загрузка

На клиентской стороне в зависимости от типа использования продукта возможны следующие варианты обращения к программе:

- в Telegram-клиенте или в веб-версии для начала работы нужно отправить чат-боту Voicee по адресу https://t.me/Voicee_AI_Bot информацию в одном из следующих форматов:
  - файл с видео/аудио;
  - ссылку на видео/аудио на облачном хранилище;
  - ссылку на видео в сервисе онлайн-видео YouTube, VK Видео или других;
  - ссылку на аудио в сервисе подкастов или прослушивания музыки;
- в интеграции с CRM не требуется вызов программы, обработка новых звонков и аудиосообщений происходит автоматически по мере их появления в CRM;
- в интеграции через API, путём отправки HTTP-запроса на адрес, выданный специалистами Voicee после подписания договора. Подробности про работу через API можно найти по адресу https://docs.voicee.ru/api/

На серверной стороне для запуска программы необходимо выполнить команду в директории, где находится docker-образ, полученной от специалистов компании «Войси» после подписания договора:

```
docker run -d -p 5000:5000 --gpus all voicee
```

После выполнения этой команды на ЭВМ, на которой она была выполнена, в случае соответствия всем требованиям к аппаратному и программному обеспечению, заработает веб-сервер, который будет отвечать на запросы на порту 5000.

В качестве альтернативы возможно выполнение команды:

```
docker compose up -d
```

В случае наличия файла docker-compose.yml в директории, поставляемой «Войси», такой способ предпочтительнее.

6 Входные данные

На клиентской стороне допускается отправка файлов форматов:

— Видеофайлы форматов MP4 (.mp4), AVI (.avi), MKV (.mkv), MOV (.mov), WMV (.wmv), FLV (.flv), WebM (.webm), MPEG (.mpg, .mpeg), 3GP (.3gp), OGV (.ogv), закодированные с помощью видео H.264 (libx264), H.265/HEVC (libx265), MPEG-4 (mpeg4), VP8 (libvpx), VP9 (libvpx-vp9), AV1 (libaom-av1), Theora (libtheora), MPEG-2 (mpeg2video), MJPEG (mjpeg), ProRes (prores) или других популярных кодеков.
— Аудиофайлы форматов MP3 (.mp3), AAC (.aac, .m4a), WAV (.wav), FLAC (.flac), OGG Vorbis (.ogg), ALAC (.m4a), AIFF (.aiff, .aif), WMA (.wma), Opus (.opus), AMR (.amr) закодированние с помощью MP3 (libmp3lame, libshine), AAC (aac, libfdkaac), FLAC (flac), Vorbis (libvorbis), Opus (libopus), ALAC (alac), PCM (pcms16le, pcms24le, pcms32le), WMA (wmav1, wmav2), AMR (libopencore-amrnb, libopencore-amrwb), Speex (libspeex) или других популярных кодеков.

На серверной стороне принимается HTTP-запрос:

```
POST /predictions HTTP/1.1
Content-Type: application/json; charset=utf-8

{
    "input": {
      "task": "transcribe",
      "debug": false,
      "gender": false,
      "warm_up": false,
      "paragraph": false,
      "batch_size": 16,
      "sentry_dsn": "",
      "diarization": false
    }
}
```

Полное описание входных параметров приводится в таблице 2.

Таблица 2 – Описание входных параметров программы

Параметр	Тип	Формат	Значение по умолчанию	Описание
audio	string	uri	-	Аудио или видео файл для обработки
cache	string	uri	-	Кэшированный файл результата в формате JSON для использования в целях отладки
task	string	-	transcribe	Выполнить транскрипцию или перевод аудиофайла

Параметр	Тип	Формат	Значение по умолчанию	Описание
vtt	string	uri	-	Файл субтитров VTT, созданный Zoom для использования в дефектоскопии
language	string	-	-	Код языка, если известен
batch_size	integer	-	16	Параллелизация обработки входных аудиофайлов
prompt	string	-	-	Начальная подсказка для распознавания речи. Используется стандартная подсказка, если не предоставлена
diarization	boolean	-	false	Определение говорящих и присвоение каждой фразе и слову
speakers	integer	-	-	Число говорящих, если известно
gender	boolean	-	false	Определение пола каждого говорящего. Работает только если диаризация включена
paragraph	boolean	-	false	Разделение текста на абзацы
debug	boolean	-	false	Вывод информации о потреблении памяти
warm_up	boolean	-	false	Возвращает пустой JSON и завершает выполнение
sentry_dsn	string	-	""	Укажите ваш Sentry DSN, если хотите собирать ошибки

7 Выходные данные

На клиентской стороне — файлы форматов PDF, TXT, DOCX или текстовые данные в сообщении Telegram с результатом работы программного комплекса Voicee.

На рисунке 2 приводится отрывок из примера выходного файла. На картинке видны подписи спикеров с обозначением пола, временные метки, текст, разбитый на абзацы, со знаками препинания и корректным разделением по предложениям.

Общество с ограниченной ответственностью «Войси»

Женский голос

01:53 Вообще бренд – это образ и эмоции. Это то, что, казалось бы, строит сама личность. То есть мы строим личный бренд, я как личный бренд его строю. Но по факту этот бренд живет где-то в умах и сердцах аудитории. То есть, по факту, это разница между тем, кто ты являешься, и между тем, как тебя воспринимают. И вот эта ценность, которая возникает в сознании или в сердце наших клиентов, является, собственно говоря, тем брендом. Это нечто неосязаемое. Это очень сложно померить, но это очень просто ощутить его наличие или отсутствие. В частности, у экспертов или у фаундеров, которые строят свои личные бренды.

Мужской голос

02:35 А как можно ощутить его наличие или отсутствие?

Женский голос

02:39 Кто-то говорит о том, что личный бренд, сильный личный бренд – это узнаваемость. И отчасти они правы.

02:46 Но узнаваемость бывает разная. Вот есть бренды медийные, то есть это бренды, у которых сотни тысяч, миллионы подписчиков или фолловеров, или тех людей, которые известны с творчеством человека или с его проектами. Ну, например, Германа Грефа знает, наверное, вся страна. От бабушки во Владивостоке до девочки в Санкт-Петербурге, которая идет снимать деньги в Сбербанк. Или предпринимателя, фаундера, который регистрируется на ПМФ, Питерский международный экономический форум. А есть бренды нишевые.

03:20 И вот что отличает один бренд? Сильный от другого – это наличие этой узнаваемости. С одной стороны, медийность является показателем сильного личного бренда, а с другой стороны, есть бренды нишевые, которые не нуждаются в такой узнаваемости, как у Германа Грефа, и отчасти это часть их стратегии. И по сути, это часть их стратегии нишевания.

Рисунок 2 – Пример выходного файла, предоставляемого пользователю программой

На серверной стороне возвращается HTTO-ответ формата:

```
HTTP/1.1 200 OK
Content-Type: application/json

{
    "status": "succeeded",
    "output": "..."
}
```

В поле output отдаётся либо ссылка скачивание файла форматов PDF, TXT, DOCX, таких же которые будут получены на клиентской стороне, либо JSON с результатами работы формата.

Перечень выходных данных представлен в таблице 3.

Таблица 3 – Описание выходных параметров программы.

Параметр	Тип	Описание
segments.text	string	Предложение, распознанное в аудио/видео

Параметр	Тип	Описание
segments.start	float	Временная метка начала предложения в аудио/видео (в секундах)
segments.end	float	Временная метка окончания предложения в аудио/видео (в секундах)
segments.words	list	Список отдельных слов в предложении и их временные метки
segments.speaker	string	Идентификатор говорящего (например, "A" или "B"), если диаризация включена
segments.paragraph	integer	Номер абзаца, к которому относится предложение (если включено разделение на абзацы)
language	string	Код языка распознанного текста
vad	list	Список блоков голосовой активности, распознаваемый период аудиофайла
vad.start	float	Временная метка начала сегмента голосовой активности
vad.end	float	Временная метка окончания сегмента голосовой активности
vad.segments	list	Вложенные сегменты в рамках блоков голосовой активности
word_segments	list	Список сегментов слов, информация о каждом распознанном слове
embeddings	dictionary	Векторные представления голосов говорящих
genders	dictionary	Информация о вероятности пола говорящих
names	dictionary	Список имён говорящих, если известен
source	dictionary	Информация об исходном файле, который был обработан: формат и длина в секундах