

ГОЛОСОВОЕ СОПРОВОЖДЕНИЕ ЛИЦ С ОВЗ ДЛЯ РАБОТЫ С ЭЛЕКТРОННОЙ ОЧЕРЕДЬЮ

*О.Б. Воскобойникова, старший преподаватель,
Ю.А. Бабенко, студентка гр. 8Е81.
Томский политехнический университет
E-mail: yab18@tpu.ru*

Введение

К категории маломобильных граждан причислены граждане с инвалидностью. Они составляют примерно 8,9 % населения, а 0,2 % из них относятся к группе населения с ограничениями по зрению. (данные Росстата РФ).

Несмотря на наличие программы «Доступная среда», положения инвалидов по зрению остаётся одним из самых затруднительных в России. Человек получает более 80 % информации посредством зрения, поэтому людям, имеющим инвалидность по зрению, приходится проходить особенно тяжёлую социальную адаптацию. Однако, современные реалии таковы, что процент улучшений либо ничтожно мал, либо не учитывает интересы самих инвалидов. В особенности это касается трудоустройства, работы социальных служб, технологического оснащения и обслуживания в бюджетных организациях. Если речь идёт об обеспечении условий для лиц с ОВЗ в сфере образования, то согласно статье 79, п. 10 федерального закона «Об образовании в Российской Федерации» образовательными организациями должны быть созданы специальные условия для обучения лиц с ОВЗ [1].

Целью работы является создание алгоритма обработки голосового запроса для процедуры получения талонов в электронной очереди для слепых и слабовидящих с помощью терминала электронной очереди.

Описание алгоритма

Для решения задачи голосового сопровождения лиц с ОВЗ с помощью систем управления очередями можно решить с использованием различных алгоритмов. В данной работе будет рассмотрен вариант с преобразованием речи в текст с использованием библиотеки Vosk. Данная библиотека основана на технологии Audio fingertipping [2].

Этап обработки полученного текста поделён на несколько составных частей:

1. Токенизация. В задачах обработки текстовой информации токенами принято называть смысловые единицы речи, т.е. слова. Соответственно, токенизация – это процесс разделения полученной строки данных на отдельные слова [4]. Данный шаг необходим для упрощения дальнейшей работы с текстом.

2. Определение члена предложения. На данном этапе для каждого из полученных токенов необходимо определить член предложения. Этот этап необходим в дальнейшем для построения подчинительной связи из полученных после предобработки слов. Определение члена предложения наиболее рационально производить при помощи классификаторов. Обычно для таких задач используют обученные классифицирующие модели, определяющие часть речи по словам, расположенным до и после текущего. Настоящая задача представляет сложность тем, что получаемые запросы преимущественно не содержат достаточного количества информации для такого метода. Классификацию решено осуществлять на основании отдельных частей слова (приставки, суффиксы, окончания) [3].

3. Лемматизация. В процессе лемматизации устраняется вариативность поисковых запросов с целью упрощения работы с данными для компьютера. На данном этапе решаются одновременно 2 задачи: слова с падежными окончаниями заменяются на слова в начальной форме и слова со схожим значением заменяются на определённый синоним, для которого в дальнейшем будет прописана определённая операция. Операция осуществляется на основе сравнения токена с базой данных синонимов определённых слов. В случае обнаружения данного слова, оно всегда заменяется на первое слово из базы данных.

4. Исключение лишних слов. На данном этапе удаляются все ненужные, «мусорные» слова, которые не несут полезной для задачи информации (например, при запросе «Мне надо получить справку» необходимо отсечь личное местоимение «мне» и слово «надо»). Отсечение можно проводить

на основании данных, полученных на втором этапе, т.е. исходя из предположения, что слова, относящиеся к определённым частям речи заведомо бесполезны.

Полученные после предобработки слова необходимо программно сравнить с заложенными в работу очереди операциями. Поскольку в задачах с использованием очереди количество возможных запросов и, соответственно, количество возможных слов ограничено, рациональным решением будет отказаться от побуквенного сравнения запроса с командой в очереди и работать с переменными, которым присвоены определённые строки с ключевыми словами. Следующий шаг необходим в ситуациях, когда данных для принятия решения не хватает. Все услуги ранжируются по популярности на основе данных о работе операторов, на этапе уточнения система выстраивает приоритет для услуг и предлагает пользователю наиболее вероятный запрос. В случае, если нужная услуга не названа, система предложит следующую наиболее популярную, пока не будет найдена искомая услуга. На основании полученного запроса в терминал очереди поступает определённый запрос, и пользователь получает талон.



Рис. 1. Визуализация алгоритма обработки запроса

Инструменты для реализации алгоритма

Для реализации алгоритма будем использовать язык программирования Python и следующие библиотеки:

- Vosk (библиотека для преобразования речи в текст);
- Pyaudio (библиотека инструментов для работы со звуковой информацией, необходима на этапе преобразования речи в текст);
- NLTK (пакет библиотек и программ для символьной и статистической обработки естественного языка, написанных на языке программирования Python, используется на этапе токенизации текста);
- String (модуль для работы с файлами);
- Rumorphy (библиотека для работы с информацией, представленной на естественном языке, используется в задачах определения части речи и лемматизации).

Заключение

Разработанный в результате выполнения работы алгоритм позволяет лицам с ОВЗ получать услуги в банках, больницах и различных муниципальных учреждениях при помощи терминала электронной очереди. Используемый подход обладает рядом следующих преимуществ:

1. Суммарное время обработки голосового запроса не будет принципиально отличаться от времени, которое тратит человек, чтобы найти в терминале необходимую услугу;

2. Не требуется больших затрат на производительные ресурсы.

Однако данный алгоритм не помогает решить некоторые проблемы. Например, задача сопровождения человека с ОВЗ после выдачи талона не решается с помощью данного алгоритма.

Список использованных источников

1. Федеральный закон «Об образовании в Российской Федерации». [Электронный ресурс]. – URL: http://www.consultant.ru/document/cons_doc_LAW_140174/ (дата обращения 12.02.2022)
2. Бабаринов С. Л. О распознавании речи / С. Л. Бабаринов, М. А. Будникова // Научные ведомости. Серия История. Политология. Экономика. Информатика. – 2014. – № 21 (192). – 182-185 с.
3. Тарасов Д.В. Процедура машинного обучения в задаче морфологической разметки текста и определения частей речи в флективных языках / Д. В. Тарасов, Н. А. Романов // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2017. – №1 (41). – 56-72 с.
4. Гречачин В.А. К вопросу о токенизации текста / В. А. Гречачин // Международный научно-исследовательский журнал – 2016. – №6 (48). – 25-27 с.