УДК 004.424 + 004.522

# Метод синхронизации аудио- и текстовой информации с применением технологии распознавания речи

В. А. Мишустин, С. В. Иваница
ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» (г. Донецк)
mishustin.post@yandex.ru

***Аннотация***
*Рассмотрены предпосылки к использованию системы распознавания речи для решения задачи синхронизации аудио- и текстовой информации. Предложен новый метод синхронизации текстовой и аудио информации – способом распознавания речи. Отмечаются особенности нового метода. Предложен программный код распознавания слов с получением временных отметок распознанных слов. Предложен алгоритм синхронизации текстовой и аудио информации. Проведено исследование, определена точность и полнота синхронизации.*

## *Введение*

Системы распознавания речи появились в целях создания устройств с голосовым управлением. Первое такое устройство было создано в 1963 году. Разработки начались в Америке и преследовали военные действия. Однако уже в 90-х годах XX столетия появились устройства способные облегчить жизнь людей с проблемами зрения.

Сейчас системы распознавания речи, в виду их повсеместного распространения (особенно в мобильных устройствах), решают прикладные задачи, например перевод речевых сигналов в текстовую информацию.

Также применяется в системах управления голосом и в интерактивных телефонных приложениях. В исследованиях [1] были предложены способы синхронизации текстовой и аудио информации. Также в данной статье предполагается, что аудио информация является речью.

Учитывая данное обстоятельство, можно предложить применение системы распознавания речи для решения задачи синхронизации текстовой и аудио информации.

## *Существующие способы синхронизации текстовой и аудио информации*

В статье [1] был проведен анализ предметной области, и предложены следующие методы синхронизации:

1. Способ процентной синхронизации, который заключается в нахождении процентного отношения текущей позиции ко всему текстовому и аудиофайлу.

2. Способ поиска уникальных наборов, который заключается в поиске искомой позиции в электронной книге путем многократной конвертации аудио потока (аудиофайл электронной книги) в текстовый формат.

3. Способ семплирования. Анализ аудио потока с целью определения (выделения) элементов текста (глава, абзац, предложение и пр.) с последующей синхронизацией с текстовым файлом одной и той же электронной книги.

Также в данной статье был проведен анализ первого предложенного метода. По результатам исследования можно заметить, что показатели точности синхронизации являются не постоянными. Несмотря на то, что погрешность находится в пределах нормы, для пользователей точность синхронизации должна быть лучше.

Предполагается, что использование системы распознавания речи позволит получить погрешность менее 0.01 секунды.

Второй и третий методы синхронизации текстовой и аудио информации, предложенные в статье [1], подразумевают анализ и обработку аудио потока. Системы распознавания речи вмещают в себя разные методы и алгоритмы по анализу и обработке аудио сигналов с целью определения речевых характеристик.

Тогда можно предложить еще один метод синхронизации текстовой и аудио информации – способом распознавания речи. Особенностью данного способа является использование существующей системы распознавания речи.

В данной работе предлагается использовать систему распознавания речи не только для распознавания речи, но и для получения отметок времени начала и конца распознаваемых слов в аудиофайле.

Таким образом, можно получить все необходимые и достаточные данные для разработки алгоритма синхронизации текстовой и аудио информации.

### *Классификация систем распознавания речи*

Обработка речевых сигналов – это область науки, в которой осуществляются фильтрация, усиление и извлечение информации, кодирование, сжатие и восстановление речи [2]. Задача обработки речевых сигналов состоит из следующих задач: нормализация, фильтрация и подавление шума; сегментация на информативные участки, определение информационных признаков, распознавание.

Проведенный обзор известных методов обработки речевых сигналов дает возможность сгруппировать методы по следующим группам:

Анализ с использованием преобразования Фурье. В области обработки речевых сигналов данный подход рассматривается как преобразование сигнала из временной в частотную область и разложение ее на частотные составляющие [3]. Это дает возможность построить спектрограмму сигнала. Однако детально анализировать локальные особенности невозможно [4].

Анализ с использованием вейвлет-преобразования. В последнее время многие задачи в области обработки речевых сигналов реализуются с использованием данного подхода [5]. Вейвлет-преобразования дают возможность анализировать кратковременные локальные особенности сигналов, путем нахождения хорошо локализованной функции как во временной, так и в частотной области.

Анализ с использованием нейронных сетей. Самым распространенным методом решения задачи распознавания речи является использование *нейронных сетей* (НС). НС представляют собой аппаратную или программную реализацию математической модели, построенной по принципу организации и функционирования биологических нейронных сетей с определенными связями между нейронами [6]. Недостатком данного метода является большая требовательность к вычислительной мощности и выполнение трудоемкой задачи обучения и подбор весовых коэффициентов синопсисов.

Анализ с использованием скрытых марковских моделей. Еще один распространенный метод распознания речевого сигнала.

Главная задача данного метода – построить статистическую модель, имитирующую работу процесса, похожего на марковский процесс с неизвестными параметрами [7].

Преимущества применения данного метода базируются на следующих предположениях [8, 9]: возможность сегментировать сигнал на фрагменты и вероятность появления символа, порожденного построенной моделью, обусловлено текущим состоянием модели.

### *Обзор существующих систем распознавания речи*

С момента появления первого устройства распознавания речи прошло более 50-ти лет. В течении этого времени данные системы развивались и находили свое применение в самых разных областях человеческой деятельности. На данный момент наибольшую популярность среди систем распознавания речи имеют: Google Speech Recognition, Microsoft Azure Speech и Microsoft Bing Voice Recognition, IBM Speech to Text.

Не смотря на большое количество преимуществ данных систем, они имеют один общие недостатки – предоставление платных услуг, и соответственно не распространяются с открытым кодом. Поэтому стоит уделить больше внимания системам, которые имеют открытый код.

Наиболее распространенными системами распознавания речи с открытым кодом являются Vosk и CMUSphinx.

Проект Vosk – это система распознавания речи, без использования сторонних ресурсов. Имеются модели для распознания более 17-ти языков и диалектов. Имеется большой функционал, однако возможность получить время начала и конца распознанного сегмента в аудиофайле отсутствует. Еще одна проблема в контексте данной работы является невозможность фильтрации шумов.

Проект CMUSphinx имеет более чем двадцатилетнюю историю развития. Помимо открытого кода и бесплатного распространения данный проект имеет ряд других преимуществ [10]: поддержка большинства языков программирования, алгоритмы распознавания речи построены на базе скрытых марковских цепях, широкий спектр дополнительных инструментов (определение ключевых слов, определение начала и конца сегмента, оценка произношения, фильтрация шумов, и другое.).

В данной работе планируется использовать систему распознавания речи, не столько для определения слов, сколько для определения отметок времени начала и конца, произнесенных слов, в аудиофайле. Проект CMUSphinx обладает наиболее подходящим функционалом.

### *Метод фрагментации текстовых элементов в аудиофайле*

Совместно с функционалом библиотеки CMUSphinx на языке программирования Python рекомендуется использовать вспомогательную библиотеку *speech_recognition*, которая обладает более обширным функционалом для дискретизации и квантования аудио сигнала.

После установки словаря фонем, акустической и языковой модели для русского языка, напишем программный код для

распознания слов из аудиофайла. Программный код представлен на рисунке 1.

При проведении данного исследования был взят отрывок (вступление автора) из аудиокниги – сказка «Сквозь зеркало и что там увидела Алиса, или Алиса в Зазеркалье» Льюиса Кэрролла. Результат работы программы представлен на рисунке 2. Программа возвращает набор строк, каждая строка содержит следующий набор данных: распознанное слово (или тишина записывается как '<*sil*>'), вероятность появления данного слова, номера сегментов на котором начинается и заканчивается распознанное слово.

```
from __future__ import print_function
import os
import speech_recognition as sr
from pocketsphinx import Pocketsphinx
import time as time

acoustic_parameters_directory = "acoustic-model"
language_model_file = "language-model.lm.bin"
phoneme_dictionary_file = "pronounciation-dictionary.dict"
start = time.time()
config = {
    'hmm': acoustic_parameters_directory,
    'lm': language_model_file,
    'dict': phoneme_dictionary_file,
    'remove_silence':False,
    'remove_noise':True,
}

ps = Pocketsphinx(**config)
AUDIO_FILE = "skazka.wav"
r = sr.Recognizer()
with sr.AudioFile(AUDIO_FILE) as source:
    audio = r.record(source)
raw_data = audio.get_raw_data(convert_rate=16000, convert_width=2)
ps.start_utt()
ps.process_raw(raw_data, False, True)
ps.end_utt()
print(ps.segments())
print('Detailed segments:', *ps.segments(detailed=True), sep='\n')
```

*Рисунок 1 – Программный код для распознавания слов из аудиофайла*

```
Detailed segments:
('<s>', -24, 0, 210)
('дитя', -938, 211, 278)
('<sil>', -5724, 279, 286)
('с', -24484, 287, 298)
('безоблачным', -20, 299, 385)
('челом', -20, 386, 450)
('<sil>', -63, 451, 456)
('ли(3)', -8567, 457, 468)
('удивлённым', -17, 469, 550)
('взглядом', -16, 551, 625)
('<sil>', -18, 626, 676)
('пусть', -28, 677, 709)
('<sil>', -3135, 710, 712)
('изменилось', -3750, 713, 796)
```

*Рисунок 2 – Пример результата работы программы*

Фрагменты тишины и вероятность появления слова, в данной работе, не представляют для нас интерес. Отфильтруем данные и оставим строки, которые содержат русские слова, и уберем из строк значение вероятности. Теперь данные представлены следующим образом – рисунок 3.

```
['дитя', 211, 278]
['с', 287, 298]
['безоблачным', 299, 385]
['челом', 386, 450]
['ли', 457, 468]
['удивлённым', 469, 550]
['взглядом', 551, 625]
['пусть', 677, 709]
['изменилось', 713, 796]
```

*Рисунок 3 – Пример отфильтрованных данных*

После фильтрации, остается 158 строк. Это значит, что система распознавания речи распознала 158 слов, и для каждого из них выделила номер сегмента начала и конца распознанного слова. Каждый сегмент имеет длительность 10 миллисекунд. То есть, чтобы получить точную отметку времени начала или конца произнесенного слова необходимо номер сегмента умножить на 10 миллисекунд. Однако текстовый вариант вступления Льюиса Кэрролла в сказке «Сквозь зеркало и что там увидела Алиса, или Алиса в Зазеркалье» имеет 159 слов. Это значит, что система распознавания речи не является идеальной системой, и не все слова были распознаны, также при первом ручном сопоставлении слов были определено, что слова могут быть неправильно распознаны.

### *Особенности синхронизации аудио- и текстовой информации*

Необходимо заметить, что данные, полученные в ходе выполнения программного кода на рисунке 1, можно воспринимать как последовательность распознанных слов, из аудиоинформации. Текстовую информацию также можно представить в виде последовательности слов.

Тогда, имея данные последовательности, задача синхронизации текстовой и аудиоинформации сводится к синхронизации двух последовательностей. Последовательность слов из текстовой информации является полной и достаточной. Последовательность распознанных слов из аудиоинформации может быть не полной, избыточной или не точной. Это объясняется не совершенностью систем распознавания речи, и как результат ошибок в распознавании речи. Тогда при синхронизации двух последовательностей необходимо учитывать, что некоторые слова из текстовой последовательности могут отсутствовать в последовательности распознанных слов, и наоборот. Тогда данные слова необходимо исключить из результирующих данных синхронизации.

То есть, задача синхронизации двух последовательностей сводится к задаче определения наибольшей общей подпоследовательности. Данная задача является классической задачей информатики и

биоинформатики, наибольшую популярность приобрел алгоритм Нидлмана-Вунша [11].

Вся суть алгоритма заключается в поэтапном заполнении матрицы, где строки представляют собой элементы первой последовательности (последовательность *x*), а колонки элементы второй последовательности (последовательность *y*). Во время прохода по матрице необходимо выполнять одно из двух действий:

1. Если элемент $x_i$ равен $y_j$, то в ячейке $(i, j)$ записывается значение ячейки $(i–1, j–1)$ с прибавлением единицы.

2. Если элемент $x_i$ не равен $y_j$, то в ячейку $(i, j)$ записывается максимум из значений $(i–1, j)$ и $(i, j–1)$.

Элементы матрицы, в которых происходило увеличение значения на единицу, необходимо добавлять в наибольшую общую подпоследовательность, при этом нужно двигаться от максимальных индексов к минимальным.

### *Применение алгоритма Нидлмана-Вунша*

Блок-схема применения алгоритма Нидлмана-Вунша для синхронизации текстовой информации и аудиоинформации представлена на рисунках 4 и 5.

Так, вначале создаем двумерный массив *audio_data* для данных полученные в ходе распознавания речи. Также необходимо создать массив *text_data* для списка слов из текста. Образованные массивы являются нашими последовательностями *x* и *y* соответственно.

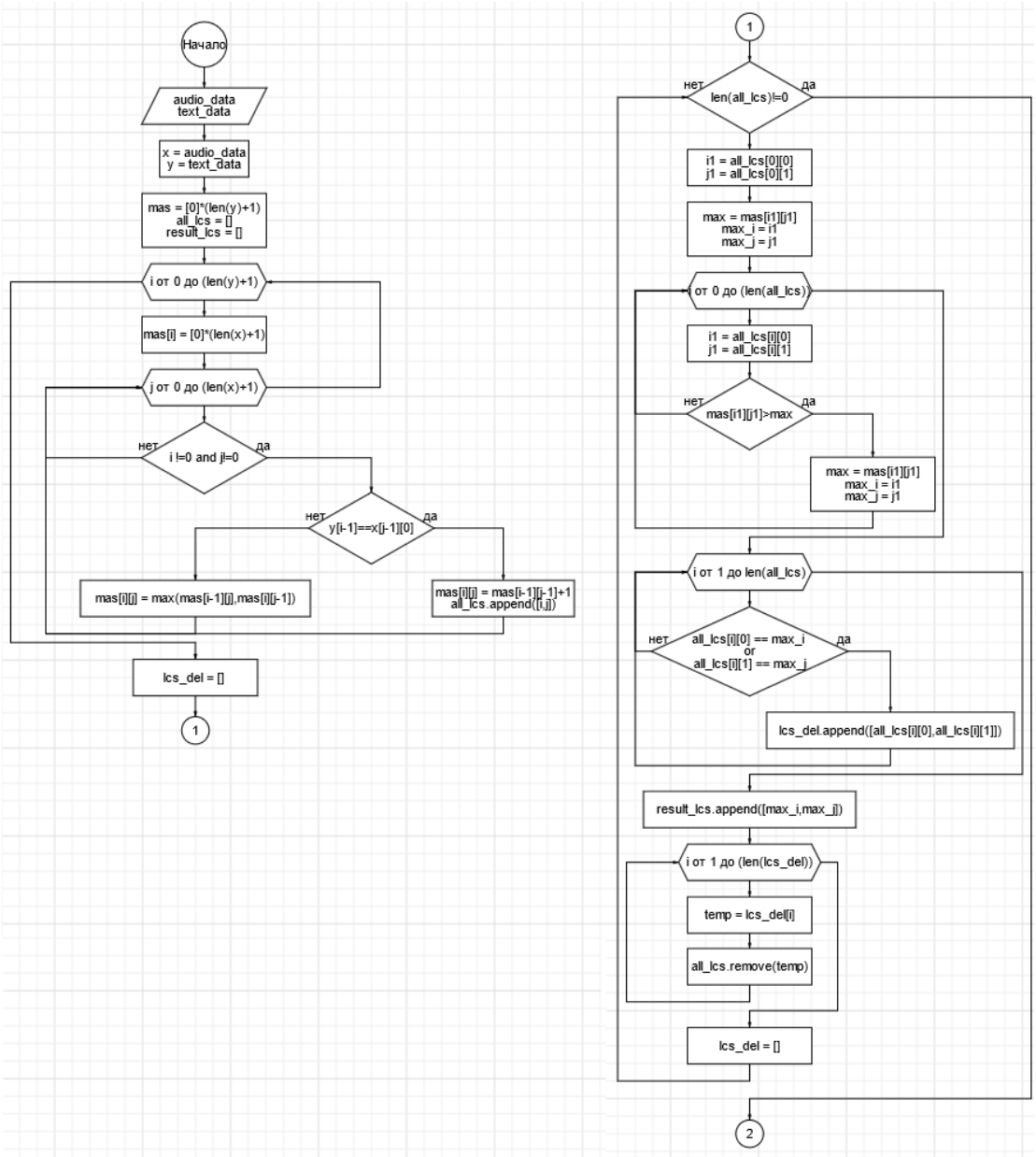


*Рисунок 4 – Блок-схема алгоритма синхронизации аудио- и текстовой информации*

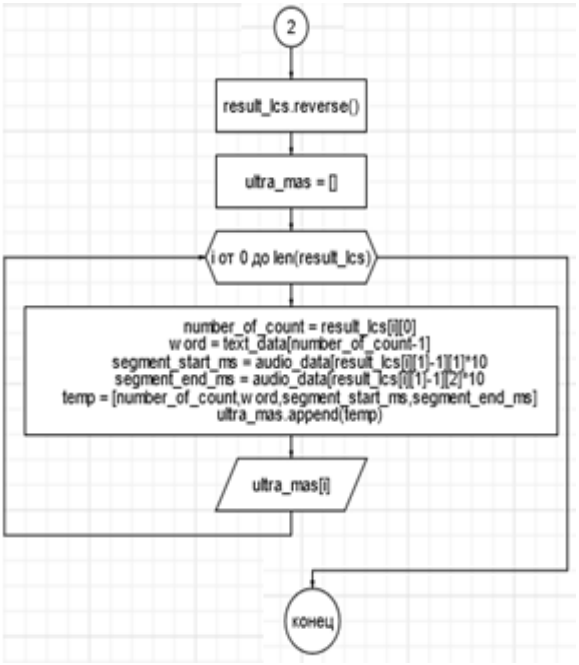


*Рисунок 5 – Блок-схема алгоритма синхронизации текстовой и аудиоинформации (продолжение блок-схемы на рисунке 4)*

После создаем двумерный массив – *mas*, размерность которого по строкам больше количества элементов в массиве *audio_data* на единицу, и по столбцам больше количества элементов в массиве *text_data* на единицу.

При использовании языка программирования Python любой созданный массив изначально заполнен нулями. При заполнении массива *mas*, строку с индексом 0 и столбец с индексом 0 не заполняем, то есть оставляем заполненными нулями.

Теперь заполняем остальной массив значениями по правилам, описанным выше. Если производится увеличение значения на единицу, тогда записываем индексы данной ячейки в ранее созданный массив *all_lcs*.

Теперь определим наибольшую общую подпоследовательность. Для этого запускаем цикл по массиву *all_lcs* по условию, пока количество элементов массива не равно нулю. Далее запускаем цикл по нахождению наибольшего значения из массива *mas*, с индексами, которые были записаны в *all_lcs*. Индексы наибольшего значения записываем в результирующий массив *result_lcs*. После из массива *all_lcs* удаляем все элементы, которые имеют любой из индексов наибольшего значения. Таким образом после того, как будет удален последний элемент из *all_lcs*, мы получим в *result_lcs* получим массив наборов индексов из массива *all_lcs*, которые входят в наибольшую общую подпоследовательность.

Первый индекс из пары значений в массиве *all_lcs* определяет слово по порядку из *text_data*. Второй индекс определяет номер сегмента по порядку, полученного в ходе распознавания речи. То есть массив является массивом синхронизации массивов *text_data* и *audio_data*. Теперь можно создать двумерный массив *ultra_mas*, который хранит все данные без ссылок на другие массивы.

Каждая строка нового массива содержит номер слова по порядку, само слово, время в миллисекундах начала и конца воспроизведения данного слова в аудиофайле. Результат работы представлен на рисунке 6.

```
Список распознанных и синхронизированных слов:
№ слова в тексте по счету, слово,
начало фрагмента (мс), конец фрагмента(мс)
Процент распознанных слов - 81.76100628930817
[1, 'дитя', 2110, 2780]
[2, 'с', 2870, 2980]
[3, 'безоблачным', 2990, 3850]
[4, 'челом', 3860, 4500]
[7, 'взглядом', 5510, 6250]
[8, 'пусть', 6770, 7090]
[9, 'изменилось', 7130, 7960]
[10, 'все', 7970, 8250]
[11, 'кругом', 8260, 8880]
[15, 'тобой', 9530, 9910]
[17, 'рядом', 10080, 10620]
[16, 'не', 46850, 47240]
```

*Рисунок 6 – Пример выходных данных*

Каждая строка результирующая строка содержит номер слова по порядку в текстовом файле, распознанное слово, время начала и конца распознанного слова в аудиофайле, в миллисекундах. В конце алгоритма проведем оценку полноты синхронизации.

## *Выводы*

В данной работе рассматриваются предпосылки к использованию систем распознавания речи для решения задачи синхронизации текстовой и аудиоинформации.

Предлагается новый метод синхронизации аудио- и текстовой информации – способом распознавания речи. Делается предположение о возможности рассматривать данные распознанных слов и слова из текстовой информации как последовательности. Алгоритм синхронизации данных последовательностей сводится к алгоритму поиска наибольшей общей подпоследовательности.

Проводится обзор алгоритма Нидлмана-Вунша. Приводится разработанный программный код реализации данного алгоритма для задачи синхронизации текстовой информации и аудиоинформации. Также приводится словесное описание и блок-схема разработанного алгоритма.

Предложенный метод предлагает абсолютную точность перехода между текстовой и аудиоинформацией.

Исследуемые в данной работе методы позволили получить полноту синхронизации с точностью совпадения, равной 82 %.

### *Литература*

1. Мишустин, В. А. Исследование способов синхронизации текстовой и аудио информации для мобильных приложений / В. А. Мишустин, С. В. Иваница // Информатика и кибернетика. – 2021. – №3(25). – С. 32–36.

2. Алимурадов, А. К. Обзор и классификация методов обработки речевых сигналов в системах распознавания речи / А. К. Алимурадов, П.П. Чураков // Измерение. Мониторинг. Управление. Конроль, 2015. – №2(12). – С. 27–35.

3. Винцюк, Т. К. Анализ, распознавание и интерпретация речевых сигналов / Т. К. Винцюк. – Киев: Наукова думка, 1987. – 264 с.

4. Рабинер, Л. Р. Цифровая обработка речевых сигналов: пер. с англ. / Л. Р. Рабинер, Р. В. Шафер. – М.: Радио и связь, 1981. – 496 с.

5. Фролов, А. В. Синтез и распознавание речи. Современные решения / А. В. Фролов, Г. В. Фролов. – М.: Связь, 2003. – 216 с.

6. Методы автоматического распознавания речи: в 2 кн.: пер. с англ. / У. А. Ли, Э. П. Нейбург, Т. Б. Мартин [и др.]; под ред. У. Ли. – М.: Мир, 1983. – Кн. 1. – 328 с.

7. Методы автоматического распознавания речи: в 2 кн.: пер. с англ. / Д. Х. Клетт, Дж. А. Барнет, М. И. Бернстейн [и др.]; под ред. У. Ли. – М.: Мир, 1983. – Кн. 2. – 392 с.

8. Моттль, В. Скрытые марковские модели в структурном анализе сигналов / В. Моттль, И. Мучник. – М.: Физматлит, 1999. – 352 с.

9. Huang, X. Spoken Language Processing. Guide to Algorithms and System Developmen / X. Huang, A. Acero, H.-W. Hon. – Prentice Hall, 2001. – 980 p.

10. Open source speech recogmition toolkit [Электронный ресурс]. – Режим доступа: https://cmusphinx.github.io/

11. Sankoff, D. Matching Sequences under Deletion/Insertion Constraints // Proc. Nat. Acad. Sci., 1972. – PP. 4–6.

***Мишустин В. А., Иваница С. В. Метод синхронизации аудио- и текстовой информации с применением технологии распознавания речи.*** *Рассмотрены предпосылки к использованию системы распознавания речи для решения задачи синхронизации аудио- и текстовой информации. Предложен новый метод синхронизации текстовой и аудио информации – способом распознавания речи. Отмечаются особенности нового метода. Предложен программный код распознавания слов с получением временных отметок распознанных слов. Предложен алгоритм синхронизации текстовой и аудио информации. Проведено исследование, определена точность и полнота синхронизации.*

***Ключевые слова:*** *система распознавания речи, текстовая и аудиоинформация, скрытые марковские цепи, алгоритм Нидлмана-Вунша.*

***Mishustin V., Ivanitsa S. Synchronization method of text and audio information using speech recognition technology.*** *The article shown the prerequisites for using a speech recognition system for solving the problem of synchronizing text and audio information. A new method of synchronizing text and audio information by using method of speech recognition is proposes. It is noted the features of the new method. Also, the article describes program code and the algorithm for synchronizing text and sound information. A study was conducted, the accuracy and completeness of synchronization were determined.*

***Keywords:*** *speech recognition system, text and audio information, hidden Markov chains, Needleman-Wunsch algorithm.*

*Статья поступила в редакцию 15.02.2022*
*Рекомендована к публикации профессором Аноприенко А.Я.*