УДК 004.522

Разработка речевого распознавателя исходного кода программ в инструментальной среде CMU Sphinx

В.С. Бакаленко
Донецкий национальный технический университет
valeriy.bakalenko@gmail.com

Бакаленко В.С. Разработка речевого распознавателя исходного кода программ в инструментальной среде CMU Sphinx. *В статье рассматривается разработка системы, предназначенной для речевого ввода и вывода исходного кода программ. Для решения данной задачи использовалась инструментальная среда CMU Sphinx-4. В результате, на основании грамматики языка программирования Pascal, были построены акустико-лингвистические модели автоматического распознавания речи и словарь.*

Ключевые слова: *речевое управление, декодер, база знаний, акустическая модель, база знаний, акустическая модель, речевой интерфейс, распознавание речи, марковские модели, словарь.*

Введение

В современном мире набор исходного кода программ выполняется вручную с помощью клавиатуры. Этот способ ввода является трудоемким и требует хороших навыков работы с клавиатурой [1]. Данный недостаток можно устранить путем качественного решения задачи автоматического распознавания речи при вводе исходного кода программ.

Задача голосового ввода текста программ заключается в распознавании лексем языка программирования. Обозначим через w – пространство образов (лексем), f(t) - функцию, которая выражает в каждый момент времени t амплитудно-частотную характеристику сигнала, g(f) – решающее правило для оценки f(t). Задача распознавания речи заключается в построении такого решающего правила g(f), которое бы позволяло проводить распознавание с минимальным числом ошибок за приемлемое время.

В решении этой задачи есть несколько основных проблем: качество распознавания, большой объем словаря, высокая скорость распознавания. Кроме того, необходимо учитывать огромное количество вариантов названия для переменных, функций и процедур. Скорость распознавания должна быть такой, чтобы программист мог в режиме реального времени видеть исходный код, который он диктует. От точности распознавания зависит количество ошибок в исходном коде и время необходимое на их устранение [2].

Поэтому цель исследования состоит в разработке и оценке эффективности речевого интерфейса, позволяющего осуществлять ввод исходного кода программ, с помощью построения акустико-лингвистических моделей, словаря и настройки на особенности голоса диктора в системе Sphinx.

Описание инструментальной среды CMU Sphinx

На сегодняшний день Sphinx является самым популярным и работоспособным из открытых движков [3]. Он разработан в университете Карнеги-Меллона с участием Массачусетского технологического института и Sun Microsystems. Достоинством Sphinx является поддержка множества языков. На его основе можно создавать свои собственные модели распознавателей речи.

Sphinx-4 является версией из семейства CMU Sphinx. Он состоит из двух компонентов: «тренера» и декодера. Тренер создаёт акустическую модель, адаптированную под конкретные потребности, а декодер выполняет собственно распознавание. Архитектура Sphinx-4 на верхнем уровне очень проста. Она включает FrontEnd, клиентскую часть (приложение), декодер и базу знаний (рис.1).

Блок FrontEnd отвечает за сбор, аннотирование и обработку входных данных. Он извлекает объекты из входных данных для чтения с помощью декодера. Аннотации определяют начальный и конечный сегменты данных. Основные операции блока FrontEnd реализуют шумоподавление, автоматическую регулировку усиления, анализ Фурье и спектральную

фильтрацию Мэла.

База знаний содержит информацию для декодера. Эта информация определяет акустическую модель и модель языка.

Декодер выполняет основную часть работы. В первую очередь он считывает данные с помощью FrontEnd и сопоставляет их с информацией из базы знаний. Затем декодер выполняет поиск в пространстве последовательностей слов, которые входят в число претендентов на выбор. Термин «пространство поиска» означает описание наиболее вероятных последовательностей слов, которые динамически обновляются с помощью декодера в процессе распознавания.

Рисунок 1 – Структура среды CMU Sphinx-4

Инструмент Sphinx-4 в отличие от других архитектур распознавателей речи предоставляет приложению контролировать некоторые функции речевого движка. Приложение во время декодирования может получать важные данные от декодера. Эти данные позволяют приложению следить за процессом декодирования, а также влиять на процесс декодирования до его завершения. Кроме того, приложение может обновлять базу знаний на любом этапе распознавания.

Главным достоинством Sphinx является возможность описания проектируемого распознавателя на уровне формальных моделей, что и послужило основанием для выбора Sphinx-4 в качестве инструментария.

Функциональная схема распознавания речи

При разработке речевого интерфейса основная проблема заключается в автоматическом распознавании речи. В данной статье она решалась на основе скрытых марковских моделей, используемых в инструментальной системе Sphinx.

Как известно, марковская модель – это вероятностный автомат с конечным числом состояний, который изменяет своё состояние один раз в единицу времени [4]. Учитывая, что марковская модель описывает вероятностные процессы, а динамические характеристики речи человека по своей природе нестабильны и не поддаются строгому математическому описанию, было принято решение построить распознаватель речи на основе аппарата скрытых марковских моделей.

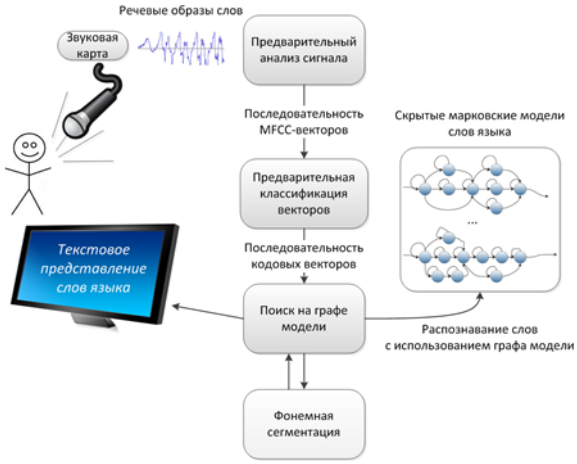


Рисунок 2 – Функциональная схема системы распознавания речи

На рисунке 2 показана функциональная схема процесса распознавания. Во время голосового ввода текста программы программист произносит лексемы языка в микрофон. Звуковая карта преобразовывает звук в цифровой сигнал. На этапе предварительной обработки сигнал преобразуется в последовательность векторов характеристик. В них выделяются фрагменты, которые соответствуют словам (лексемам). Каждое слово разбивается на фонемы и им сопоставляются наиболее вероятные состояния скрытой марковской модели. В итоге марковская модель для каждого входного речевого образа определяет текстовое изображение слова.

Рассмотрим пример скрытой марковской модели (рис. 3). Слева и справа изображена одна и та же модель. Слева обозначены состояния и переходы между ними, справа – наблюдаемые символы и тоже переходы между ними. В данном случае количество состояний N=3, количество наблюдаемых символов M=3 (A, B, C). Количество состояний может быть больше количества наблюдаемых символов, т.к. с одним и тем же символом могут быть связаны разные состояния. Матрица переходов между состояниями A представляет собой таблицу размером 3 на 3. Матрица вероятностей появления символов наблюдения B имеет аналогичный размер.

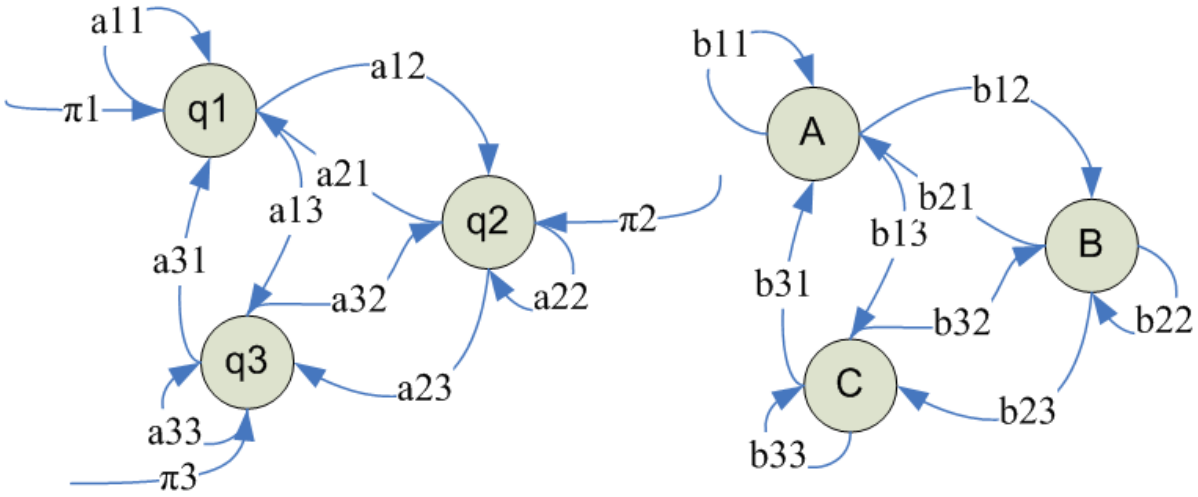


Рисунок 3 – Скрытая марковская модель с 3 состояниями

В распознавании речи используются лево-правые скрытые марковские модели. Их последовательность состояний обладает свойством, которое выражается в том, что с увеличением времени индекс состояния также увеличивается или же остается неизменным. Т.е. состояния переходят слева направо, а наоборот сделать переход нельзя. Этот тип отлично подходит для описания процессов с прямым ходом времени, в частности, для распознавания речевых сигналов.

Акустико-лингвистическая модель системы распознавания лексем языка

В первую очередь технология Sphinx предусматривает разработку акустико-лингвистической модели языка [5]. В неё входят: словарь с транскрипциями, грамматика и обучающее множество для акустической модели. Составляющие акустической модели показаны на рисунке 4.

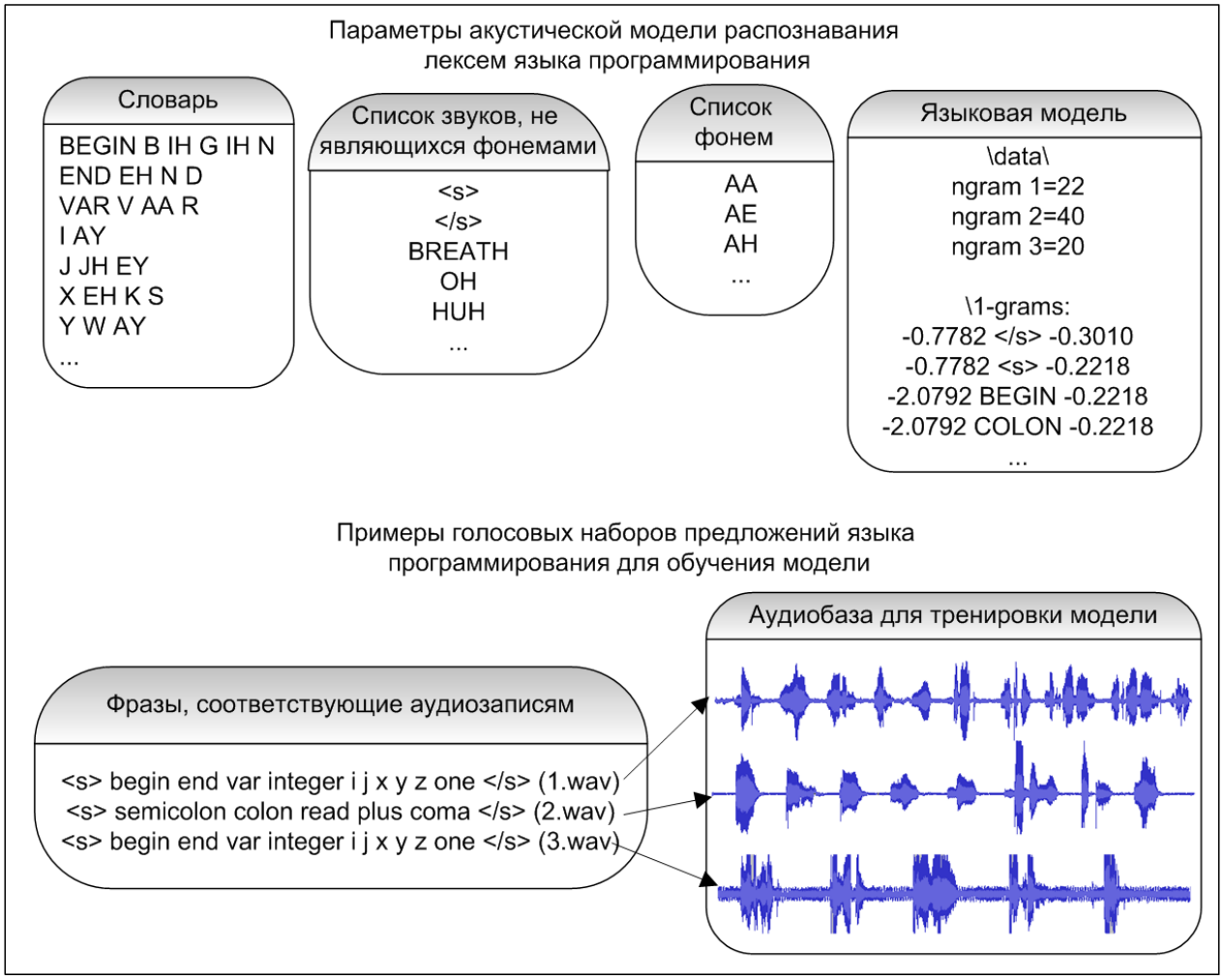
Словарь содержит список слов и транскрипции к ним. Транскрипции должны состоять исключительно из фонем, которые присутствуют в списке фонем. Кроме слов учитываются и другие побочные звуки: звуки дыхания, различный шум окружающей среды, шум от звуковой аппаратуры.

Языковая модель – это совокупность вероятностей появления слов в речи.

Для обучения модуля распознавания записывается аудиобаза. Она представляется в виде обучающих примеров с вариантами речи конкретного диктора. Каждая аудиозапись должна иметь своё текстовое представление. От объёма и содержания материала аудиобазы зависит качество распознавания. Элементы информационной структуры лингвистической модели показаны на рис. 5.

По введенным характеристикам лингвистической модели распознаваемого языка инструмент Sphinx формирует акустическую модель [1,2]. Она является ядром интерфейса речевого ввода исходного кода программ. Акустическая модель состоит из скрытых марковских моделей, необходимых для распознавания речи. Для обучения скрытой марковской модели инструмент Sphinx использует алгоритм Баума-Велша [6]. Акустическая модель содержит вероятности появления кластеров, которые объединяются в фонемы. Наиболее вероятное слово определяется по распознанным фонемам, транскрипциям в словаре и вероятностям появления слов из языковой модели [7-10].

Рисунок 4 – Структура акустической модели



3 Рисунок 5 – Информационная структура лингвистической модели

Анализ качества построенных моделей

Окно пользовательского интерфейса системы речевого ввода программ разработанное для тестирования построенных моделей показано на рисунке 6.

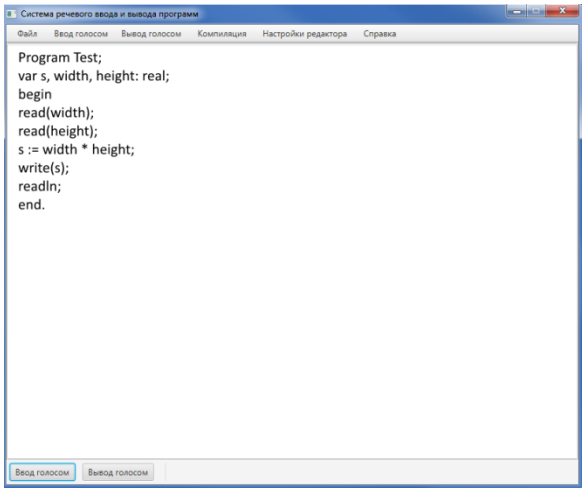


Рисунок 6 – Интерфейс речевой системы ввода исходного кода программ

Лучшие показатели работы модуля распознавания речи были тогда, когда он был настроен на дикторозависимое распознавание.

Для настройки речевого интерфейса под конкретного диктора строились соответствующие обучающие множества. Всего в тестовой программе было 144 слова, и аудиозапись длилась 123,67 секунд. Каждая из акустических моделей проходила ряд тестов с использованием двух языковых моделей. Первая – модель, построенная на словах, вторая – по часто встречающимся словосочетаниям.

Первый тест проводился с акустической моделью, обученной на 100 аудиофайлах. На этой маленькой модели ясно видно, что при малом количестве обучающего материала, распознавание будет сильно зависеть от качества языковой модели. При плохой языковой модели процент

ошибки составил 48,96%, а при хорошей – 37,44%.

Второй тест проводился с акустической моделью, обученной на 293 аудиофайлах. На этой модели также видна зависимость качества распознавания от языковой модели. При плохой модели ошибка составила 12,96%, а при хорошей – 2,88%.

Третий тест проводился с акустической моделью, обученной на 500 аудиофайлах и на этом тесте уже не так заметна разница в качестве распознавания в зависимости от различных типов моделей. При плохой модели ошибка составила 7,2 %, а при хорошей – 2,88%. Это может сказать о том, что чем больше имеется обучающего материала, тем меньше распознавание зависит от качества языковой модели. Результаты всех этих тестов изображены на рисунке 7.

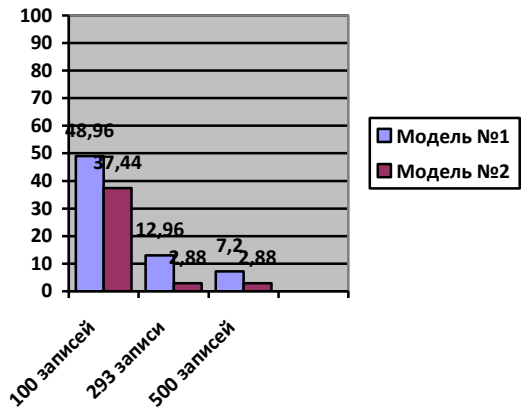


Рисунок 7 - Результаты тестирования акустических моделей с разными языковыми моделями, процент ошибки

Одним из важных аспектов распознавания речи является оценка скорости распознавания речи и количество потребляемой оперативной памяти.

Количество потребляемой памяти явно не зависит от качества моделей и от количества обучающего материала. Всего приложению было выделено 400мб оперативной памяти и во всех случаях количество потребляемой разрабатываемым программным продуктом памяти приближалось к этому порогу.

Заключение

Проблемы в области создания приложений с речевым вводом не имеют однозначных и тривиальных решений.

По показателю точности распознавания можно сделать следующие выводы:

- однодикторная акустическая модель имеет точность распознавания выше, чем дикторонезависимая;
- с увеличением объёма словаря ухудшается качество распознавания;
- использование грамматики не ухудшает точность распознавания;
- автоматная грамматика даёт лучшие результаты распознавания, чем контекстно-свободная из-за маленького размера контекста;
- грамматика исключает появление лишних лексем при наличии шума.

По критерию скорости распознавания можно сделать следующий вывод: автоматная грамматика работает медленнее контекстно-свободной из-за большего количества правил. Длительность распознавания одного слова составляла половину времени его произношения, что является приемлемым результатом.

Речевой интерфейс с дополнительными сервисными функциями предоставит начинающим программистам более естественный способ набора исходного кода программы.

Литература

1. Ли У.А. и др. Методы автоматического распознавания речи: В 2-х книгах. Пер. с англ. / Под ред. У. Ли. М.: Мир, 1983. - Кн. 1. 328 е., ил.

2. Рабинер Л.Р., Шафер Р.В. Цифровая обработка речевых сигналов: Пер. с англ. / Под ред. М.В. Назарова и Ю.Н. Прохорова. М.: Радио и связь, 1981.

3. Савченко В.В., Акатьев Д.Ю., Губочкин И.В. Формирование фонетической базы данных из речевого сигнала на основе информационной теории восприятия речи. // Системы управления и информационные технологии. 2008. 4.1 (34). С. 193-198.

4. Блеихут Р. Быстрые алгоритмы цифровой обработки сигналов: Пер. с англ.- М.: Мир, 2002.

5. Елинек Ф. Распознавание непрерывной речи статистическими методами//ТИИЭР. 1976. Т. 64. №4. С. 131-160.

6. Чучупал В. Я. Выделение незнакомых слов и акустических событий при распознавании речи // Модели, методы, алгоритмы и архитектуры системраспознавания речи, 2006, стр. 119-137.115

7. Моттлъ В.В., Мучник И.Б. Скрытые марковские модели в структурном анализе сигналов. М.: Физматлит, 1999, 352 с.

8. Yu, D. Automatic Speech Recognition: A Deep Learning Approach [Text] /

D. Yu, L. Deng. — London : Springer-Verlag, 2015.

9. Bourlard, H. Continuous speech recognition using multilayer perceptrons with hidden Markov models [Text] / H. Bourlard, C. Wellekens // Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). — 1990. — P. 413—416.

10. Morgan, N. Neural networks for statistical recognition of continuous speech [Text] / N. Morgan, H. Bourlard // Proc. IEEE. — 1995. — Vol. 83, no. 5. — P. 742—772.

Бакаленко В.С. Разработка речевого распознавателя исходного кода программ в инструментальной среде CMU Sphinx. *В статье рассматривается разработка системы, предназначенной для речевого ввода и вывода исходного кода программ. Для решения данной задачи использовалась инструментальная среда CMU Sphinx-4. В результате, на основании грамматики языка программирования Pascal, были построены акустико-лингвистические модели автоматического распознавания речи и словарь.*

Ключевые слова: *речевое управление, декодер, база знаний, акустическая модель, база знаний, акустическая модель, речевой интерфейс, распознавание речи, марковские модели, словарь.*

Bakalenko V.S. Development of a speech recognition of the source code of programs in the CMU Sphinx tool environment. *The article deals with the development of a system designed for speech input and output of the source code of programs. To solve this problem, the CMU Sphinx-4 tool environment was used. As a result, based on the grammar of the Pascal programming language, acoustic-linguistic models of automatic speech recognition and a dictionary were built.*

Keywords: *speech control, decoder, knowledge base, acoustic model, knowledge base, acoustic model, speech interface, speech recognition, Markov models, dictionary.*

Статья поступила в редакцию 20.03.2018
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом