УДК 004.522

КОМПЛЕКС ПРОГРАММНЫХ СРЕДСТВ ОБРАБОТКИ И РАСПОЗНАВАНИЯ РАЗГОВОРНОЙ РУССКОЙ РЕЧИ

И. С. Кипяткова,[1]
младший научный сотрудник
Санкт-Петербургский институт информатики и автоматизации РАН

Представлен программный комплекс для обработки и распознавания разговорной русской речи. В него входят блоки обучения моделей акустических единиц речи, предварительной обработки текстового материала, создания транскрипций слов, выбора лучших транскрипций, создания n-граммной модели русского языка, автоматического распознавания речи. Дается детальное описание программных модулей, входящих в каждый из этих блоков.

Ключевые слова *— распознавание, русская речь, обучение, системы распознавания речи, n-граммная модель языка, автоматическое фонематическое транскрибирование.*

Введение

Автоматическое распознавание разговорной русской речи представляет собой очень сложную задачу и по сравнению с распознаванием изолированных слов требует дополнительных программных модулей. Во-первых, произношение слов в разговорной речи сильно варьируется, и фонетическое представление произнесенных слов зачастую не совпадает с транскрипциями слов, сделанными по фонетическим правилам транскрибирования. Поэтому кроме программного модуля создания базовых фонематических транскрипций необходим модуль генерации альтернативных транскрипций, которые учитывали бы различные варианты произнесения слов в разговорной речи. Во-вторых, при автоматическом распознавании разговорной слитной речи распознавателю необходима модель языка, описывающая допустимые фразы. Однако в русском языке отсутствуют жесткие грамматические конструкции предложений, что затрудняет создание моделей языка.

Архитектура программных средств обработки разговорной русской речи представлена на рис. 1. Система работает в двух режимах — обучение и распознавание. В режиме обучения создаются модели акустических единиц речи, модель языка, а также словарь словоформ, которые далее будут использоваться распознавателем. Для обучения акустических моделей используется вручную размеченный корпус русской речи, а модель языка создается по текстовому корпусу. Таким образом, можно выделить следующие этапы обучения системы распознавания:

— предварительная обработка текстового материала для создания модели языка;

— создание транскрипций для слов из собранного текстового корпуса;

— выбор наилучших транскрипций;

— создание модели языка;

— обучение акустических единиц речи.

В режиме распознавания входной речевой сигнал преобразуется в последовательность векторов признаков, и затем производится поиск наиболее вероятной гипотезы с использованием предварительно обученных акустических и языковых моделей.

Ниже будут подробно описаны программные модули, реализующие эти режимы работы. Программные модули созданы на языках программирования C++ и Perl, а также используются сторонние модули в виде исполняемых файлов, в том числе модули комплексов программ CMU-Cambridge Statistical Language Modeling Toolkit (CMU SLM) [1], HTK (Hidden Markov Model Toolkit) [2], АОТ (Автоматическая обработка текста) [3].

Представленная архитектура программных средств для обработки русской речи составляет основу системы автоматического распознавания

[1] Научный руководитель — кандидат технических наук, старший научный сотрудник СПИИРАН *А. А. Карпов*.

■ ***Рис. 1.*** *Архитектура программных средств обработки разговорной русской речи*

и понимания речи SIRIUS (SPIIRAS Interface for Recognition and Integral Understanding of Speech), разработанной в СПИИРАН.

Блок обучения моделей акустических единиц речи

Обучение моделей акустических единиц речи производится с использованием корпуса русской речи. Для обеспечения дикторонезависимости системы распознавания требуются речевые базы данных с записями большого числа дикторов. Запись производится в шумоизолированной комнате. Диктору последовательно показываются фразы, которые следует произнести, и каждая его фраза записывается в отдельный звуковой wav-файл. После записи речевых данных выполняется полуавтоматическая разметка акустического сигнала на фразы, слова и фонемы.

Обучение моделей акустических единиц осуществляется следующим образом. Вначале происходит сегментация речевого сигнала — разделение на короткие сегменты длительностью 10–30 мс, примерно соответствующие квазистационарным участкам речи. Для более точного описания сигнала речевые сегменты обычно берутся с перекрытием. Каждый раз рассматривается лишь один короткий сегмент сигнала, а остальная часть сигнала отбрасывается. В спектральной области такая процедура соответствует операции свертки спектра участка сигнала с помощью нелинейной функции окна, для того чтобы ослабить краевые эффекты на границах сегментов. Обычно для этих целей используют окно Хэмминга [4].

Затем, речевые данные, проходя уровень параметрического представления, посегментно преобразуются в последовательность векторов признаков. Для параметрического представления речевого сигнала производится спектральный анализ сегментов сигнала с вычислением кепстральных коэффициентов по мел-шкале частот (Mel Frequency Cepstral Coefficient — MFCC) с их первой и второй производными [5].

Для акустического моделирования речи используются скрытые марковские модели (СММ), при этом каждая фонема (звук речи) представляется одной непрерывной СММ первого порядка. Модель фонемы имеет три состояния: первое описывает начало фонемы, второе представляет центральную часть и третье — концовку. СММ слова получается путем соединения в цепочку моделей фонем из соответствующего фонетического алфавита. Аналогичным образом соединяются модели слов друг с другом, образуя модели фраз. Цель обучения акустических моделей, основанных на СММ, состоит в том, чтобы по обучающей последовательности наблюдений определить такие параметры модели, с которыми вероятность появления этой последовательности была максимальной [6]. Одним из способов выбора таких параметров модели, чтобы локально максимизировать данную вероятность, является метод Баума—Уэлча. В ходе процесса обучения определяется распределение вероятностей переходов между состояниями СММ (матрица переходных вероятностей), распределение вероятностей появления символов наблюдения в состоянии, вероятностное распределение начальных состояний модели. Итеративно переобучаются данные параметры моделей, пока происходит увеличение вероятности распознавания обучающего набора фраз.

На выходе данного блока формируется набор СММ акустических единиц речи для принятого фонетического алфавита.

Блок предварительной обработки текстового материала

Обучающий текстовый материал собирается с интернет-сайтов электронных газет (например, www.ng.ru («Независимая газета»), www.smi.ru («СМИ.ru»)), откуда закачиваются html-файлы. Каждый из этих файлов обрабатывается модулем HTMLrem, который удаляет в них теги и преобразует их в текстовые файлы. Затем получившиеся текстовые файлы объединяются в один, при этом содержащийся в них текст разбивается на отдельные предложения, а каждое предложение записывается с новой строки. На части также делятся предложения, содержащие прямую и косвенную речь, кроме того, точка с запятой считается границей раздела двух предложений.

Модуль нормализации текста (cleanup.pl) выполняет автоматическую обработку собранного текстового материала, которая включает в себя удаление повторяющихся предложений; удаление текста, написанного в скобках; удаление предложений, состоящих из пяти и меньшего количества слов; расшифровку общепринятых сокращений; удаление знаков препинания; замену заглавной буквы на строчную в словах, начинающихся с заглавной буквы [7].

Модуль (text2wfreq) создания частотного словаря является модулем CMU. На вход модуля поступает текст, а на выходе создается частотный словарь, который представляет собой список всех словоформ, используемых в тексте, с частотой их встречаемости в тексте.

Следующий модуль (wfreq2vocab) из комплекса программ CMU создает из частотного словаря алфавитный словарь словоформ, используемых в тексте. Размер словаря можно задать двумя способами: напрямую указав максимальный размер

словаря либо указав максимальный порог частоты появления слова, начиная с которой слова будут записываться в словарь.

Таким образом, в результате работы блока предварительной обработки текстового материала создается нормализованный обучающий текстовый корпус, а также частотный и алфавитный словари словоформ из данного корпуса.

Блок создания транскрипций слов

Транскрипции для слов из собранного текстового корпуса генерируются автоматически с помощью программных модулей создания транскрипций. Вручную должны быть созданы транскрипции для аббревиатур, а также для некоторых широко распространенных слов, заимствованных из английского языка.

Первоначальным этапом создания транскрипций является определение ударной гласной/гласных в слове. Для этого используется модуль простановки ударений (GetUdarenieWord) с применением базы данных словоформ русского языка, созданной путем объединения двух баз данных, свободно доступных в Интернете: морфологической базы данных проекта STARLING [8] и морфологической базы данных проекта AOT [9]. Для слов, которые отсутствуют в этой базе данных, а также для сложных слов, у которых в этой базе данных отсутствует информация о второстепенном ударении, модуль пытается определить ударение автоматически. Автоматически ударение ставится над *ё*, а также если в слове только одна гласная. Если обрабатываемое слово содержит дефис, а в базе данных для него отмечено только одно ударение, то это слово разбивается на две части, и затем они по отдельности проверяются по базе данных ударений. Если они обнаруживаются в базе данных, то второстепенное ударение ставится на первое слово, а основное — на второе. Подробнее работа данного модуля описана в статье [10].

Модуль создания базовых транскрипций (TranscribeWord) создает фонематические транскрипции слов. На вход модуля поступает алфавитный словарь словоформ из текстового корпуса, для которых транскрипции создаются с использованием базовых фонетических правил [11] и модуля GetUdarenieWord. В качестве фонетического алфавита используется модифицированный вариант международного фонетического алфавита SAMPA. В нашем варианте используются 48 фонем: 12 — для гласных звуков (с учетом ударных вариантов) и 36 — для согласных (с учетом твердости и мягкости звуков). Алгоритм создания базовых транскрипций слов описан в работе [12].

Модуль создания транскрипций для аббревиатур (abbrev.pl) автоматически создает транскрипции для некоторых аббревиатур. Существует три типа прочтения аббревиатур: буквенный, звуковой и буквенно-звуковой [13]. При буквенном типе прочтения транскрипции создаются автоматически, если транскрипция состоит только из согласных. Тогда буквы слова заменяются на их звуковое прочтение, а ударение ставится на последний гласный в слове. В случае звукового типа прочтения транскрипции создаются автоматически для аббревиатур вида: согласный—гласный—согласный. При этом транскрипция создается аналогично как и для других слов. Аббревиатуры с буквенно-звуковым типом прочтения автоматически не обрабатываются, так как в них не очевиден вариант правильного произношения.

Завершающий модуль создания альтернативных транскрипций (AdvancedTranscribator) из списка базовых транскрипций слов и аббревиатур создает альтернативные транскрипции, которые учитывают вариативность произношения слов в разговорной речи. Разработанный модуль фонематического транскрибирования создает альтернативные транскрипции, используя правила, описывающие возможные явления редукции и ассимиляции фонем [14, 15]. Процесс создания альтернативных транскрипций детально описан в статье [12].

В результате работы данного блока создается словарь базовых и альтернативных транскрипций словоформ, соответствующий собранному текстовому корпусу.

Блок отбора наилучших транскрипций

Модуль принудительного выравнивания транскрипций HVite из комплекса программ HTK выбирает из сгенерированных альтернативных транскрипций одну наиболее подходящую соответствующему речевому сигналу из базы данных. В этом случае выбор транскрипции происходит только между альтернативными транскрипциями одного и того же слова [16]. Для каждой фразы алгоритм Витерби вычисляет вероятность того, что фонематическая транскрипция и речевой сигнал соответствуют друг другу [17]. Наибольшие вероятности при выравнивании транскрипций каждого слова позволяют выбрать оптимальные варианты [16].

Модуль подсчета частоты выбора альтернативных транскрипций (FrequencyFromAligned) определяет, сколько раз каждая транскрипция была выбрана при выполнении принудительного выравнивания, и создает частотный словарь транскрипций слов.

Следующий модуль выбора часто употребляемых транскрипций по порогу (CreateNewDictBasedOnFrequency) из частотного словаря транскрипций выбирает те, частота появления кото-

рых больше заданного порога, и эти транскрипции добавляются к базовым в словарь системы распознавания.

В результате работы блоков создания транскрипций и выбора лучших альтернативных транскрипций создается список фонематических представлений слов из текстового корпуса. В этот список входят базовые транскрипции, а также наилучшие альтернативные для тех слов из текстового корпуса, которые присутствовали в обучающем корпусе речи.

Блок создания *n*-граммной модели языка

Задача *n*-граммной модели языка состоит в оценке вероятности появления цепочки слов в некотором тексте (например, в гипотезе распознавания фразы). *n*-граммы представляют собой последовательность из *n* элементов (например, слов), а *n*-граммная модель языка используется для предсказания элемента в последовательности, содержащей $n - 1$ предшественников. Модуль создания списка *n*-грамм (text2idngram) входит в состав CMU. Входными данными являются обучающий текстовый корпус и алфавитный словарь слов из этого текста. В результате работы модуля создается список *n*-грамм с частотой их появления в обучающем корпусе. Модуль позволяет создавать *n*-граммы с различным значением *n*, но на практике обычно используются биграммы ($n = 2$), которые определяют вероятность появления пар слов, и триграммы ($n = 3$), которые определяют вероятность появления троек слов.

Ввиду нежесткого порядка слов в русском языке многие грамматически связанные пары слов оказываются в предложении разделены другими словами, и в результате статистического анализа текста не создаются биграммы, содержащие такие пары слов, а *n*-граммные модели языка оказываются недостаточно эффективными. Увеличить количество создаваемых в результате обработки текста различных *n*-грамм и тем самым повысить качество модели языка позволит выявление грамматически связанных пар слов за счет синтаксического анализа предложений. Поэтому в данный блок программ для создания модели языка встроен модуль синтаксического анализа VisualSynan проекта АОТ. Целью синтаксического анализа является построение синтаксических групп в предложении [3]. Синтаксическую группу определяют следующие параметры: тип группы, пара синтаксически связанных слов, граммемы. Тип групп — это строковая константа: ПРИЛ_СУЩ (прилагательное—существительное), ПГ (предложная группа) и т. д. Граммемы группы — это морфологические характеристики слов, которые определяют поведение и сочетаемость элементов в других группах. Поскольку синтаксическая группа представляет собой пару слов, то синтаксический анализ можно применить только при создании биграммной модели языка.

Модуль получения *n*-грамм в результате синтаксического анализа (syntax_parse.pl) выполняет обработку результатов, полученных в ходе синтаксического анализа. Суть обработки заключается в следующем: грамматически связанные пары слов (синтаксические группы) добавляются в список биграмм, если в тексте они разделены другими словами. В список биграмм добавляются следующие синтаксические группы: прилагательное—существительное (например, классическое и авангардное направление), подлежащее—сказуемое (мы ее знали), прямое дополнение (отмечали свое десятилетие), наречие—глагол (уже полностью распределены), генитивная пара (темой данного номера), отсравнительная группа (важнее самого контракта), глагол—инфинитив (придется сначала попробовать), обособленное прилагательное в постпозиции (цель, достаточно благородную), причастие—существительное (горы, снегами наполненные), существительное—придаточное определительное предложение (отзывы, которые приходят).

Следующий модуль объединения списков *n*-грамм (merge_bigr.pl) объединяет списки биграмм, созданные в результате как статистической обработки биграмм, так и синтаксического анализа текста.

Модуль удаления *n*-грамм с частотой появления, меньшей заданного порога, (cut.pl) сокращает список *n*-грамм путем удаления тех элементов, частота появления которых меньше заданного порога. На вход модуля подается частотный список *n*-грамм и задается порог, на выходе получается сокращенный список *n*-грамм.

Далее модуль удаления *n*-грамм со словами, отсутствующими в словаре транскрипций, (ngram_final.pl) проводит анализ, для каких слов из получившегося списка *n*-грамм были ранее созданы транскрипции. *n*-граммы со словами, для которых транскрипции не были созданы, удаляются этим модулем.

Из-за удаления некоторых *n*-грамм из модели языка появляются слова, которые в модели не приводят к конечному результату (разрывают цепочку слов), поскольку встречаются в *n*-граммах не во всех позициях. Модуль удаления *n*-грамм со словами, присутствующими не во всех позициях, (del.pl) отфильтровывает *n*-граммы, содержащие такие слова.

Модуль создания модели языка в формате ARPA (idngram2lm) является модулем CMU. Этот модуль по списку *n*-грамм создает вероятностную модель языка. Модель языка в формате ARPA

а)

```
\data\
ngram 1=208183
ngram 2=6013323

\1-grams:
…
-5.9467 абонемент      -0.3327
-6.1268 абонемента     -0.1235
-6.6619 абонементе     -0.0696
-7.2060 абонементном   -0.3172
-7.3821 абонементную   -0.3358
-6.2517 абонементов    -0.1658
-6.3120 абонементы     -0.2449
…

\2-grams:
…
-0.7822 абонемент </s>
-1.9771 абонемент БСО
-0.9235 абонемент в
-1.9771 абонемент где
-1.9771 абонемент геннадия
-1.9771 абонемент для
-1.9771 абонемент и
-1.7042 абонемент из
…
```

б)

```
VERSION=1.0
N=208183        L=6013323
…
I=1312  W=абонемент
I=1313  W=абонемента
I=1314  W=абонементе
I=1315  W=абонементном
I=1316  W=абонементную
I=1317  W=абонементов
I=1318  W=абонементы
…
J=138694        S=1312  E=0     l=-0.7822
J=138695        S=1312  E=178   l=-1.9771
J=138696        S=1312  E=18579 l=-0.9235
J=138697        S=1312  E=32837 l=-1.9771
J=138698        S=1312  E=33228 l=-1.9771
J=138699        S=1312  E=43482 l=-1.9771
J=138700        S=1312  E=57416 l=-1.9771
J=138701        S=1312  E=58025 l=-1.7042
…
```

■ ***Рис. 2.*** *Фрагмент представления модели языка в формате ARPA (а) и SLF (б)*

имеет вид, показанный на рис. 2, *а*. Вначале в модели языка идет список униграмм, слева от униграммы указывается значение десятичного логарифма вероятности ее появления, справа — коэффициент возврата (back-off weight) [18], который применяется в тех случаях, когда некоторая n-грамма отсутствует в обучающем корпусе или частота ее появления очень низкая, тогда вместо нее используется вероятность (n – 1)-граммы, умноженная на коэффициент возврата. Ниже идет список биграмм с вероятностями их появления.

Модуль (lmconvert.pl) перевода модели языка в формат SLF (Standard Lattice Format) трансформирует модель языка из формата ARPA в более компактный формат, применяемый в HTK. Фрагмент модели языка в формате SLF представлен на рис. 2, *б*, где I — порядковый номер словоформы, W — словоформа, J — порядковый номер биграммы, S — порядковый номер первого слова в биграмме, E — порядковый номер последнего слова в биграмме, l — десятичный логарифм вероятности появления биграммы в тексте.

Результатом работы блока создания модели языка является n-граммная статистическая модель языка, отражающая данные из обучающего текстового корпуса.

Блок распознавания речи

Первым этапом процесса распознавания речи является определение границ речи в звуковом сигнале, поступающем от микрофона. Для этого используется свойство отличия значений энергии для речевых сегментов сигнала и для сегментов фонового шума. Затем осуществляется сегментация речевого сигнала и извлечение признаков. Эти этапы являются аналогичными этапам процесса обучения моделей акустических единиц речи.

Для распознавания слитной речи используется модифицированный алгоритм Витерби, называемый методом передачи маркеров (token passing method) [2]. Метод передачи маркеров определяет прохождение возможных путей по состояниям объединенной СММ. В начало каждого слова из словаря ставится маркер и применяется итеративный алгоритм оптимизации Витерби, при этом на каждом шаге сдвигается маркер и для него вычисляется вероятностная оценка по акустической и языковой модели. После обработки всей последовательности векторов наблюдений выбирается маркер, имеющий наибольшую вероятность. Когда наилучший маркер достигает конца обрабатываемого сигнала (последовательности наблюдений), то путь, которым он проходит через сеть, известен в виде истории (хранящейся в маркере), и из маркера считывается последовательность пройденных слов, которая и является гипотезой распознавания фразы. Кроме того, может быть получено несколько наилучших маркеров, таким образом создается список лучших гипотез произнесенной фразы (N-best list). В дальнейшем этот список гипотез может быть обработан для выбора одной наилучшей гипотезы на основе синтаксического, семантического или прагматического анализа.

Заключение

Представленная архитектура программных средств для обработки русской речи составляет основу системы автоматического распознавания и понимания речи SIRIUS. Встроенный в программный комплекс модуль создания альтернативных транскрипций, учитывающих явления возможной редукции и ассимиляции звуков речи, позволяет использовать данную систему для распознавания разговорной речи. Применение синтаксического анализа при создании модели языка позволяет выявлять грамматически связанные пары слов, которые были разделены в тексте другими словами, что повышает эффективность биграммных моделей. В зависимости от задачи в качестве обучающего текстового корпуса могут использоваться тексты из различных предметных областей, таким образом возможно получить предметно-ориентированную систему автоматического распознавания речи.

Работа выполняется при поддержке Минобрнауки РФ в рамках ФЦП «Кадры» (госконтракты № П2579 и П2360), Совета по грантам Президента РФ (проект № МК-64898.2010.8) и фонда РФФИ (проект № 11-08-01016-а).

Литература

1. **Clarkson P., Rosenfeld R.** Statistical language modeling using the CMU-Cambridge toolkit // Proc. EUROSPEECH. Rhodes, Greece, 1997. P. 2707–2710.
2. **Young S.** et al. The HTK Book (for HTK Version 3.4). — Cambridge, UK, 2009. — 375 p.
3. **Сокирко А. В.** Морфологические модули на сайте www.aot.ru // Диалог-2004. Компьютерная лингвистика и интеллектуальные технологии: Тр. Междунар. конф. М.: Наука, 2004. С. 559–564.
4. **Rabiner L., Juang B.-H.** Fundamentals of Speech Recognition. — Prentice Hall, 1995. — 507 p.
5. **Strom N.** Continuous Speech Recognition in the WAXHOLM Dialogue System // Stockholm QPSR. 1996. P. 67–95.
6. **Ронжин А. Л., Карпов А. А., Ли И. В.** Речевой и многомодальный интерфейсы. — М.: Наука, 2006. — 173 с. (Информатика: неограниченные возможности и возможные ограничения).
7. **Кипяткова И. С., Карпов А. А.** Автоматическая обработка и статистический анализ новостного текстового корпуса для модели языка системы распознавания русской речи // Информационно-управляющие системы. 2010. № 4(47). С. 2–8.
8. **Проект** «Эволюция языка». Русские словари и морфология. http://starling.rinet.ru/morpho.php?lan=ru (дата обращения: 24.03.2011).
9. **Автоматическая** обработка текста. Исходники словарей и программ. http://www.aot.ru/download.php (дата обращения: 24.03.2011).
10. **Кипяткова И. С., Карпов А. А.** Разработка и оценивание модуля транскрибирования для распознавания и синтеза русской речи // Искусственный интеллект. Донецк, Украина, 2009. № 3. С. 178–185.
11. **Русская** грамматика: В 2 т. / Редкол.: Н. Ю. Шведова (гл. ред.) и др. — М.: Наука, 1980. — 783 с.
12. **Кипяткова И. С., Карпов А. А.** Модуль фонематического транскрибирования для системы распознавания разговорной русской речи // Искусственный интеллект. 2008. № 4. С. 747–757.
13. **Кривнова О. Ф.** Обработка инициальных аббревиатур при автоматическом синтезе речи // Тр. Междунар. семинара по компьютерной лингвистике и ее приложениям «Диалог99». М., 1999. http://www.philol.msu.ru/~otipl/SpeechGroup/publications/kriv_di2.doc (дата обращения: 24.03.2011).
14. **Лобанов Б. М., Цирульник Л. И.** Моделирование внутрисловных и межсловных фонетико-акустических явлений полного и разговорного стилей в системе синтеза речи по тексту // Анализ разговорной русской речи (АР^3 — 2007): Тр. первого междисциплинарного семинара. СПб.: ГУАП, 2007. С. 57–71.
15. **Русская** разговорная речь / Под ред. Е. А. Земской. — М.: Наука, 1973. — 485 с.
16. **Kessens J. M., Wester M., Strik H.** Improving the performance of Dutch CSR by modeling within-word and cross-word pronunciation variation // Speech Communication. 1999. Vol. 29. P. 193–207.
17. **Saraclar M.** Pronunciation Modeling for Conversational Speech Recognition: PhD thesis. — Baltimore, USA, 2000. — 143 p.
18. **Moore G. L.** Adaptive Statistical Class-based Language Modelling: PhD thesis. — Cambridge University, 2001. — 193 p.