Математические структуры и моделирование
2011, вып. 24, с. 41–54

УДК 004.934.8'1

ОБЗОР ОСНОВНЫХ МЕТОДОВ РАСПОЗНАВАНИЯ ДИКТОРОВ

Е.А. Первушин

В статье представлен обзор методов, используемых для решения задачи распознавания по голосу. Уделяется внимание устоявшейся структуре систем распознавания. Также приводятся краткие описания наиболее распространённых методов извлечения признаков (таких как MFCC и LPCC), а также методов классификации (метод векторного квантования, модель гауссовых смесей, метод опорных векторов). Обсуждаются методы оценки систем распознавания и представления результатов таких оценок.

Введение

Задача распознавания дикторов является актуальной задачей области речевых технологий. Связь распознавания дикторов с остальными областями обработки речи может быть отражена следующими отношениями (по классификации в [6]).

Обработки речи может быть разбита на задачи
- анализа/синтеза,
- распознавания,
- кодирования.

Распознавание охватывает подзадачи
- распознавания речи,
- распознавания дикторов,
- идентификации языка.

Распознавание дикторов объединяет идентификацию и верификацию дикторов.

Идентификация диктора — процесс определения личности по образцу голоса путём сравнения данного образца с шаблонами, сохранёнными в базе. Результатом процесса идентификации является список кандидатов. Реализующая система может выдавать список фиксированного размера либо принимать решение о включении пользователя в список кандидатов на основании заданного порога. Если предусмотрена возможность того, что в процессе идентификации будет участвовать пользователь, не зарегистрированный в системе, то говорят

Омский государственный университет им. Ф.М. Достоевского
E-mail: pervushin_evgen@mail.ru

об идентификации на открытом множестве. В идеальном случае для такого пользователя система должна выдать пустой список. Если все пользователи, проходящие процедуру идентификации, зарегистрированы в системе, то говорят об идентификации на замкнутом множестве.

Верификация диктора — процесс, при котором с помощью сравнения представленного образца с хранимым в базе шаблоном проверяется запрошенная идентичность. Результатом верификации является положительное либо отрицательное решение. Иногда используется термин «обнаружение по голосу» (speaker detection [8]). В задаче обнаружения используются несколько иные термины и приоритеты, но, по сути, верификация и обнаружение являются одной той же задачей.

Помимо данной классификации сами системы распознавания могут быть разделены на текстозависимые и текстонезависимые в зависимости от того, известен ли системе текст, который должен быть произнесён пользователем, и использует ли система данную информацию. При текстозависимом распознавании могут использоваться как фиксированные фразы, так и фразы, сгенерированные системой и предложенные пользователю. Текстонезависимые системы предназначены обрабатывать произвольную речь.

Существуют и другие задачи, связанные с распознаванием по голосу. К числу таких можно отнести следующую задачу. Пусть сигнал содержит запись разговора двух или более лиц. Часть сигнала размечается вручную или с использованием алгоритмов обучения без учителя для указания, кто в какой момент говорит. Остальная часть сигнала должна быть размечена автоматически. В этой задаче помимо идентификации фрагментов требуется определить также их границы. Такая задача получила в англоязычной литературе название speaker diarization («протоколирование дикторов» в [1]).

1. Структура систем распознавания дикторов

Работа систем распознавания содержит два основных этапа: регистрация пользователей в системе и сам процесс распознавания (попытка идентификации или верификации). Пользователи предварительно регистрируются в системе, записав свои голоса. Образец голоса каждого диктора обрабатывается с целью извлечения признаков, которые могут быть использованы для распознавания. На основе извлечённых признаков строятся модели (в некоторых случаях более подходящим термином является «шаблон») пользователей. Модель представляет собой некоторую структуру, позволяющую при данных признаках оценить степень подобия либо сразу принять решение.

В случае верификации пользователь пытается войти в систему, предъявляя идентификатор и образец голоса. Признаки, извлечённые из предъявленного образца, сравниваются с соответствующей моделью, сохранённой в базе, а также, возможно, с референтной моделью, представляющей фиксированное множество некоторых пользователей, либо наиболее близких к данному голосу. Результат сравнивается с заданным порогом и выдаётся положительное или отрицательное решение о допуске.

Во время процесса идентификации также происходит извлечение признаков из предъявленного образца, которые затем сравниваются с моделями всех зарегистрированных в системе пользователей либо предварительно отобранных.

Таким образом, общая схема системы распознавания реализуется с помощью следующих основных этапов или уровней.

- Уровень обработки сигналов. На данном уровне сигнал обрабатывается с целью выделить признаки, существенные для задачи распознавания. Речевой сигнал представляется с помощью последовательности векторов признаков.
- Уровень моделей. При регистрации пользователя данный уровень использует полученную от уровня обработки сигналов последовательность векторов признаков для построения модели. Моделирование может заключаться как в простом копировании векторов признаков, так и в построении вероятностных моделей или других структур. После чего становится возможным при данных признаках вычислить степень подобия между признаками и сохранённой моделью.
- Уровень принятия решений. Функции принятия решений традиционно выделяют в отдельный уровень, хотя он может выполнять тривиальные функции или отсутствовать, если на уровне моделей вычисляются конечные решения. Для принятия решений используются степени подобия, вычисленные на уровне моделей, и, если необходимо, заданные пороги.

2. Получение образца и его предобработка

При распознавании по голосу обрабатываемым образцом является запись речевого сигнала. При кодировании импульсно-кодовой модуляцией аналоговый сигнал представляется последовательностью мгновенных измерений значений амплитуд (отсчётов). Для записи и обработки речевого сигнала обычно используется частота дискретизации 8 или 16 кГц, более высокая частота дискретизации требует больших вычислительных расходов. Для представления отсчётов используется 8, 12 или 16 бит, также допустимы другие значения.

На качество распознавания влияет ряд факторов, связанных с записью и передачей речевого сигнала. Среди них можно выделить следующие:

- несовпадение канала,
- плохая акустика помещения,
- различное расстояние до микрофона и прочее.

Например, при использовании распознавания голоса, передаваемого по телефонному каналу, в общем случае нельзя гарантировать использование для регистрации и идентификации одного и того же микрофона и канала передачи, кроме того необходимо учитывать влияние посторонних помех. Использование более качественной записи возможно, например, в приложении верификации по голосу для контроля доступа к помещению. В таком случае канал представляет собой микрофон, его кабель и аналого-цифровой преобразователь.

Предварительная обработка сигнала может заключаться в удалении участков, не содержащих речь, а также обработке частотным фильтром.

3. Методы извлечения признаков

Обработка сигнала в данных приложениях имеет целью выделить в речевом сигнале информацию, релевантную для задачи распознавания по голосу, то есть информацию, представляющую индивидуальные особенности голоса человека, или признаки. Выделенные признаки будут использованы для формирования шаблона или для сравнения с зарегистрированными шаблонами. Априори невозможно оценить, какие признаки более подходят для распознавания. Процесс определения подходящих признаков заключается в переборе возможных вариантов признаков с последующей экспериментальной оценкой.

Выделяют два вида признаков: низкоуровневые (обусловленные анатомическим строением речевого аппарата) и высокоуровневые (приобретённые, связанные с манерой произношения).

Сложившийся подход к процедуре обработки речевого сигнала состоит в использовании кратковременного анализа. То есть сигнал разбивается на временные окна фиксированного размера, на которых, как предполагается, параметры сигнала не меняются. Для речевого сигнала размер окна обычно выбирается в пределах 10–30 мс. Для более точного представления сигнала между окнами делают перекрытие, равное половине длины окна. Затем к каждому окну применяются алгоритмы извлечения признаков, такие как спектральный анализ, метод линейного предсказания или другие.

3.1. Мэл-частотные кепстральные коэффициенты

Данный метод извлечения признаков является одним из самых распространённых как в системах распознавания дикторов, так и в системах распознавания речи.

На вход алгоритма подаётся последовательность отсчётов участка сигнала, исследуемого на данной итерации, $x_0, \ldots, x_{N-1}$. К данной последовательности применяется весовая функция и затем дискретное преобразование Фурье. Весовая функция используется для уменьшения искажений в Фурье анализе, вызванных конечностью выборки. На практике в качестве весовой функции часто используется окно Хэммига, которое имеет следующий вид:

$$w_n = 0{,}54 - 0{,}46 \cdot \cos\left(2\pi \frac{n}{N-1}\right), \; n = 0, \ldots, N-1,$$

где N — длина окна, выраженная в отсчётах.

Тогда дискретное преобразование Фурье взвешенного сигнала можно записать в виде

$$X_k = \sum_{n=0}^{N-1} x_n w_n e^{\frac{-2\pi i}{N} kn}, \; k = 0, \ldots, N-1.$$

Значения индексов k соответствуют частотам

$$f_k = \frac{F_s}{N} k, \; k = 0, ..., N/2,$$

где F_s — частота дискретизации сигнала.

Полученное представление сигнала в частотной области разбивают на диапазоны с помощью банка (гребёнки) треугольных фильтров. Границы фильтров рассчитывают в шкале *мэл*. Данная шкала является результатом исследований по способности человеческого уха к восприятию звуков на различных частотах. Перевод в мэл-частотную область осуществляют по формуле

$$B(f) = 1127 \cdot \ln\left(1 + \frac{f}{700}\right).$$

Обратное преобразование выражается как

$$B^{-1}(b) = 700\left(e^{b/1127} - 1\right).$$

Пусть N_{FB} — количество фильтров (обычно используют порядка 24 фильтров), (f_{low}, f_{high}) — исследуемый диапазон частот. Тогда данный диапазон переводят в шкалу мэл, разбивают на N_{FB} равномерно распределённых перекрывающихся диапазона и вычисляют соответствующие границы в области линейных частот. Обозначим через $H_{m,k}$ — весовые коэффициенты полученных фильтров. Фильтры применяются к квадратам модулей коэффициентов преобразования Фурье. Полученные значения логарифмируются

$$e_m = \ln\left(\sum_{k=0}^{N} |X_k|^2 H_{m,k}\right),\ m = 0, \ldots, N_{FB} - 1.$$

Заключительным этапом в вычислении MFCC коэффициентов является дискретное косинусное преобразование

$$c_i = \sum_{m=0}^{N_{FB}-1} e_m \cos\left(\frac{\pi i(m + 0{,}5)}{N_{FB}}\right),\ i = 1, \ldots, N_{MFCC}.$$

Коэффициент c_0 не используется, так как представляет энергию сигнала. Количество коэффициентов N_{MFCC} на практике выбирают порядка 12.

3.2. Кепстральные коэффициенты на основе линейного предсказания

Суть линейного предсказания заключается в том, что линейной комбинацией некоторого количества предшествующих отсчётов можно аппроксимировать текущий отсчёт

$$x_n \approx \sum_{k=1}^{p} a_k x_{n-k}.$$

Весовые коэффициенты линейной комбинации $a_1, \ldots, a_p$ называются коэффициентами линейного предсказания.

Нахождение коэффициентов линейного предсказания осуществляют с помощью рекурсивного алгоритма Дарбина [2].

На основе полученных коэффициентов линейного предсказания рассчитываются кепстральные коэффициенты. Причём таких коэффициентов может быть сгенерировано больше, чем самих коэффициентов линейного предсказания

$$c_n = \begin{cases} a_n + \sum\limits_{k=1}^{n-1} \dfrac{k}{n} c_k a_{n-k}, & 1 \leqslant n \leqslant p; \\ \sum\limits_{k=n-p}^{n-1} \dfrac{k}{n} c_k a_{n-k}, & n > p. \end{cases}$$

Для сигнала с частотой дискретизации 8000 Гц используют порядка 12 коэффициентов линейного предсказания, из которых генерируют порядка 18 кепстральных коэффициентов.

4. Обработка признаков

Описанные выше методы извлечения признаков предназначены для выделения характеристик на небольшом участке. Для того чтобы сохранить информацию о динамике речи, применяют подход, заключающийся в объединении векторов признаков с их первыми и, возможно, вторыми производными. Такие производные получили название Δ- и Δ-Δ- коэффициентов (дельта- и дельта-дельта-коэффициентов).

На этапе постобработки признаков также применяют методы нормализации, использующие весь набор векторов признаков исследуемой записи. Наиболее распространённым методом нормализации, предназначенным для снижения влияния канала, является метод вычитания кепстрального среднего (Cepstral Mean Substraction; CMS). Данный метод предназначен для компенсации изменений между сессиями и в применении к постоянным условиям, наоборот, снижает эффективность.

5. Методы классификации

Распознавание по голосу отличается от многих биометрических систем тем, что в данном случае предметом распознавания является процесс, а не статическое изображение, как в случае с распознаванием отпечатков пальцев, лица или радужной оболочки глаза. Поэтому чаще всего образец голоса представляется не в виде единого вектора признаков, а в виде последовательности векторов признаков, каждый из которых описывает характеристики небольшого участка речевого сигнала. Последовательность векторов, полученная после этапа обработки сигнала, используется для построения шаблона/модели диктора или для осуществления сравнения с уже построенными шаблонами. Для задач верификации и идентификации может быть определён способ вычисления степеней подобия предъявленного образца с одним или несколькими шаблонами. Степень подобия может вычисляться на основе определённой метрики или на основе оценки вероятности.

Существует несколько способов классификации моделей для задачи распознавания. В литературе часто ссылаются на модели как на генеративные или дискриминативные. Суть моделей, которые называют генеративными, заключается в моделировании данных, полученных для обучения, например, с помощью оценки функции плотности вероятности. Примером может служить модель гауссовых смесей. Дискриминативные модели основаны на построении границы между классами, как это реализовано, например, в методе опорных векторов.

5.1. Вычисление расстояния

Определение метода вычисления расстояния является основой для шаблонных моделей. В таких моделях распознаваемый объект рассматривается как неточная копия одного из хранимых.

Одними из самых распространённых методов вычисления расстояния между векторами являются следующие:

- L1-норма (расстояние городских кварталов, манхэттэнское расстояние)

$$d(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{D} |x_i - y_i|;$$

- евклидово расстояние

$$d(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{D} (x_i - y_i)^2;$$

- расстояние Махалонобиса

$$d(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^T \times W^{-1} \times (\mathbf{x} - \mathbf{y}),$$

где W — матрица ковариации. В случае если W равна единичной матрице, расстояние совпадает с евклидовым.

5.2. Метод ближайшего соседа

В качестве шаблона диктора в данном методе используется полный набор векторов обучающей последовательности. Сравнение образца с таким шаблоном происходит следующим образом. Каждый вектор тестовой последовательности сравнивается с каждым вектором шаблона для определения минимального расстояния. Полученные расстояния усредняются для формирования итоговой оценки

$$d_k = \sum_{i=1}^{L} \min_{\mathbf{x}_j \in Sp_k} d(\mathbf{x}_i, \mathbf{x}_j).$$

Для снижения вычислительной трудоёмкости используют различные методы сокращения шаблона либо методы сохранения данных для ускорения поиска, такие, как, например, kd-дерево или другие методы [5].

Метод k-ближайших соседей также предписывает сохранение последовательности обучающих векторов, однако вычисление степени подобия происходит несколько иным способом. Для каждого тестового вектора $\mathbf{y}_i$ после вычисления расстояний до векторов хранимых шаблонов могут быть найдены k ближайших векторов. Пусть k_{ij} – количество векторов среди найденных k ближайших, принадлежащие классу j (диктору j в нашем случае). Предполагая более или менее одинаковое количество векторов обучения в каждом классе, оценка вероятности принадлежности вектора i классу j может быть получена как

$$\hat{P}(C_j|\mathbf{y}_i) = \frac{k_{ij}}{k}.$$

Тогда последовательность векторов может быть классифицирована по правилу

$$C = \arg\max_{1 \leqslant j \leqslant N} \prod_{i=1}^{L} \hat{P}(C_j|\mathbf{y}_i).$$

Вместо данного правила (именуемого иногда «правилом произведения») с целью сглаживания эффекта, производимого выбросами, обладающими нулевой или близкой к нулю оценкой вероятности, вводят «правило суммы»

$$C = \arg\max_{1 \leqslant j \leqslant N} \sum_{i=1}^{L} \hat{P}(C_j|\mathbf{y}_i).$$

Используя формулу оценки вероятности и учитывая тот факт, что количество соседей k является постоянным, формула может быть переписана как

$$C = \arg\max_{1 \leqslant j \leqslant N} \sum_{i=1}^{L} k_{ij}.$$

Такую технику называют схемой голосования, так как последовательность классифицируется к классу, набравшему наибольшее количество «голосов» [10].

5.3. Векторное квантование

В методе векторного квантования в отличие от метода ближайшего соседа множество обучающих векторов сохраняется не целиком, а преобразуется в множество (обычно фиксированного размера) кодовых векторов. Распространённым методом построения такого множества, именуемого также кодовой книгой, является алгоритм K-средних.

Алгоритм K-средних разбивает исходное множество на K кластеров, где K — предварительно заданное число. Для этого сначала значения средних инициализируются некоторыми векторами из исходного множества. Затем на каждой итерации алгоритма происходит распределение векторов в ближайшие к ним кластеры (для этого вычисляется расстояние между вектором и текущими значениями средних) и перерасчёт среднего в каждом кластере. Алгоритм

завершается после того, как на очередной итерации состояния кластеров не изменились либо по достижении заданного максимального количества итераций. Полученные значения средних являются кодовыми векторами, используемыми для построения шаблона. Вычисление расстояния между входной последовательностью векторов и кодовыми книгами осуществляется аналогично методу ближайших соседей.

5.4. Модель гауссовых смесей

Модель гауссовых смесей широко используется в области распознавания дикторов. Данная модель представляет собой взвешенную сумму Гауссиан

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{M} w_i p_i(\mathbf{x}),$$

где λ — модель диктора, M — количество компонентов модели, w_i — веса компонентов такие, что

$$\sum_{i=1}^{M} w_i = 1.$$

Функция плотности вероятности каждого компонента даётся формулой

$$p_i(\mathbf{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu_i)^T\Sigma_i^{-1}(\mathbf{x}-\mu_i)\right),$$

где D — размерность пространства признаков, μ_i — вектор математического ожидания, Σ — матрица ковариации. Чаще всего в системах, реализующих данную модель, используется диагональная матрица ковариации. Возможно также использование одной матрицы ковариации для всех компонентов модели диктора или одной матрицы для всех моделей.

Таким образом, для построения модели диктора необходимо определить векторы средних, матрицы ковариации и веса компонентов. Данную задачу решают с помощью EM-алгоритма. На вход подаётся обучающая последовательность векторов $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$. Параметры модели инициализируются начальными значениями и затем на каждой итерации алгоритма происходит переоценка параметров.

Для определения начальных параметров обычно используют алгоритм кластеризации такой, как алгоритм K-средних [3]. Построив разбиение множества обучающих векторов на M кластеров, параметры модели могут быть инициализированы следующим образом. Начальные значения μ_i совпадают с центрами кластеров, матрицы ковариации рассчитываются на основе попавших в данный кластер векторов, веса компонентов определяются долей векторов данного кластера среди общего количества обучающих векторов.

Переоценка параметров происходит по следующим формулам:

- вычисление апостериорных вероятностей (Estimation-step)

$$p(i|\mathbf{x}_t, \lambda) = \frac{w_i p_i(\mathbf{x}_t)}{\sum_{k=1}^{M} w_k p_k(\mathbf{x}_t)};$$

- вычисление новых параметров модели (Maximization-step)

$$w_i = \frac{1}{T}\sum_{t=1}^{T} p(i|\mathbf{x}_t, \lambda); \qquad \mu_i = \frac{\sum_{t=1}^{T} p(i|\mathbf{x}_t, \lambda)\mathbf{x}_t}{\sum_{t=1}^{T} p(i|\mathbf{x}_t, \lambda)};$$

$$\Sigma_i = \frac{\sum_{t=1}^{T} p(i|\mathbf{x}_t, \lambda)(\mathbf{x}_t - \mu_i)(\mathbf{x}_t - \mu_i)^T}{\sum_{t=1}^{T} p(i|\mathbf{x}_t, \lambda)}.$$

Данные шаги повторяются до схождения параметров.

5.5. Метод опорных векторов

Метод опорных векторов является бинарным классификатором и строит разделяющую функцию в виде

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b.$$

Пусть дана обучающая последовательность $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$, где $\mathbf{x}_i$ — точки пространства признаков, y_i — метки, обозначающие принадлежность одному из классов, принимающие значения 1 или -1. Рассмотрим пока случай линейной разделимости данных. Такое ограничение может быть записано в виде

$$\begin{cases} \mathbf{w} \cdot \mathbf{x}_i + b \geqslant +1, & \mathbf{y}_i = +1; \\ \mathbf{w} \cdot \mathbf{x}_i + b \leqslant -1, & \mathbf{y}_i = -1; \end{cases} \tag{1}$$

или одним неравенством

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 \geqslant 0, \ i = 1, \ldots, N.$$

Среди возможных разделяющих гиперплоскостей ищется гиперплоскость, создающая максимальный зазор между классами. То есть расстояние от разделяющей гиперплоскости до ближайших точек каждого класса максимально. Величина зазора может быть посчитана как $2/\|\mathbf{w}\|$. Тогда задачу поиска разделяющей гиперплоскости с максимальным зазором удобно свести к минимизации $\|\mathbf{w}\|^2$ при условиях 1. Данная задача (так же как описываемые далее обобщения на случай линейной неразделимости и введения нелинейности с

помощью функции ядра) может быть решена методами квадратичного программирования. Одним из самых популярных является метод, предложенный в [11].

Для того чтобы обобщить задачу на случай линейной неразделимости, ограничения переписывают в виде

$$\begin{cases} \mathbf{w} \cdot \mathbf{x}_i + b \geqslant +1 - \xi_i, \ \mathbf{y}_i = +1 \\ \mathbf{w} \cdot \mathbf{x}_i + b \leqslant -1 + \xi_i, \ \mathbf{y}_i = -1 \\ \xi_i \geqslant 0 \end{cases}$$

для всех $i = 1, \ldots, N$. Целевая функция принимает вид

$$\frac{1}{2}\mathbf{w} \cdot \mathbf{w} + C \sum_{i=1}^{N} \xi_i \to \min,$$

где C — положительная постоянная, задающая степень штрафа за появление ошибок.

Другим способом, позволяющим распознавать линейно-неразделимые множества, является введение функции ядра. Идея заключается в том, чтобы отобразить исходное пространство в пространство более высокой размерности, в котором, как может оказаться, множества разделимы. Причём, поскольку всюду в алгоритмах обучения и распознавания признаки используются не отдельно, а в виде скалярных произведений, то нет необходимости в явном виде строить такое преобразование. Достаточно задать функцию ядра, определяющую скалярное произведение в новом пространстве

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j).$$

Среди распространённых можно привести следующие ядра:

$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\dfrac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$ — ядро радиальных базисных функций Гаусса,

$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + 1)^n$ — полиномиальное ядро.

Параметры метода (такие как C и параметры ядра) обычно определяют с помощью перебора некоторого множества значений и оценкой методом кросс-валидации.

Описанные выше методы решают задачу бинарной классификации. Для применения данных методов к задаче многоклассового распознавания используют такие стратегии, как «один-против-остальных» или «один-против-одного». Пусть для обучения получены данные q классов. При использовании стратегии «один-против-остальных» создаются q классификаторов, каждый из которых обучается отличать данный класс от всех остальных. При распознавании объект приписывается к тому классу, чей классификатор выдал наибольшее значение функции $f(\mathbf{x})$. Стратегия «один-против-одного» («каждый против каждого») использует $q(q-1)/2$ классификаторов, разделяющих по два класса.

Результаты сравнений конкретного класса с каждым из остальных суммируются и затем сравниваются с аналогичными других классов. При этом используют также различные способы преобразования функции $f(\mathbf{x})$ в вероятность $P(C_i|\mathbf{x})$.

Для распознавания последовательности векторов признаков могут быть применены правила, комбинирующие результаты классификации каждого кадра. Используются также подходы, при которых последовательности векторов используются для обучения генеративных моделей, которые затем классифицируются с помощью метода опорных векторов.

6. Оценка систем

На точность работы систем распознавания оказывает влияние ряд факторов. Прежде всего необходимо отметить изменчивость самого голоса. Эмоциональное состояние, усталость, возрастные изменения, простуда и многие другие факторы влияют на голос. Во-вторых, проблемой для систем распознавания является влияние окружающей среды, а также изменение условий записи.

Базы данных (корпуса), используемые для экспериментальной оценки, не всегда способны смоделировать перечисленные ситуации. Поэтому результат существенно зависит от того, насколько представительна база и как построен эксперимент. Для того чтобы получить представление об адекватности эксперимента реальным условиям применения, а также получить возможность сравнивать результаты, исследователи приводят детальную информацию о проведённых опытах. Такая информация, во-первых, должна содержать указание количества сессий записи и продолжительность интервалов между ними. Во-вторых, описание условий записи (тип микрофона, канал передачи, зашумлённость помещения и прочее) и являются ли условия различными для сессий регистрации и тестовых сессий (mismatched conditions). Результат также зависит от продолжительности материала, используемого в каждом тесте и для создания моделей, и от количества пользователей в базе.

Для оценки систем идентификации в большинстве случаев ограничиваются замкнутым множеством пользователей, то есть все пользователи, проходящие попытку идентификации, зарегистрированы в системе. Результат зависит от количества зарегистрированных пользователей и от размера возвращаемого списка (чаще всего используют только один идентификатор) или от порога включения в список. Вероятность идентификации (истинно-положительной идентификации) оценивают как долю попыток идентификации, в результате которых был возвращён список кандидатов, содержащий верный идентификатор.

В системах верификации возникают ошибки двух типов. Ошибка ложного допуска заключается в принятии положительного решения (о сходстве) при сравнении шаблонов двух разных пользователей. Принятие решения о различии образцов в то время, как они принадлежат одному пользователю, называют ошибкой ложного недопуска. Оба типа ошибок зависят от порога принятия решений.

Результаты испытаний верификации могут быть представлены с помощью графика рабочей характеристики, представляющую собой параметрически за-

данную кривую порога принятия решений. На таком графике по оси абсцисс откладываются оценки вероятностей ложно-положительных решений (вероятности ложного допуска), а по оси ординат — оценки вероятностей истинно-положительных решений. В области верификации дикторов приобрёл популярность несколько иной способ, при котором по оси абсцисс откладываются оценки вероятностей ложного допуска, а по оси ординат — оценки вероятностей ложного недопуска. При этом для большей наглядности для осей используют шкалу нормального отклонения [9] (иногда логарифмическую или иную шкалу). Такой способ предложен в [7] и был использован для представления результатов в задаче обнаружения дикторов (speaker detection), поэтому получил название кривой компромисса ошибок обнаружения (Detection Error Tradeoff; DET).

Для того чтобы представить результаты оценок в виде единого параметра, используют следующие способы. Один из них состоит в задании стоимостей ложного допуска (C_{FA}) и ложного недопуска (C_{FR}) и вычислении функции стоимости обнаружения (Detection Cost Function; DCF) [8]

$$DCF = C_{FR}P_{tar}R_{FR} + C_{FA}P_{imp}R_{FA},$$

где P_{tar} и P_{imp} — априорные вероятности попыток подлинного лица и «самозванца», R_{FR} и R_{FA} — полученные оценки вероятностей ошибок ложного недопуска и ложного допуска соответственно. Порог принятия решений оптимизируется, чтобы минимизировать значение функции стоимости. Более популярной мерой является уровень равной вероятности ошибок (Equal Error Rate; EER), который представляет величину вероятности ошибок при таком пороге, при котором вероятности ошибок ложного допуска и ложного недопуска совпадают или наиболее близки по значению.

Заключение

Развитие систем распознавания дикторов осуществляется по нескольким направлениям.

Наиболее распространёнными методами извлечения признаков являются методы вычисления кепстральных коэффициентов: мэл-частотных и на основе линейного предсказания. Также применяют статистики основного тона и формантные частоты [4]. Развитие на уровне обработки сигнала происходит, в основном, в направлении поиска новых методов обработки сигнала, имеющих целью робастное представление речевого сигнала, то есть устойчивое к внешним шумам и искажениям, вызванным каналом передачи сигнала. Развивается направление, в котором осуществляется использование высокоуровневых признаков.

Методы создания моделей дикторов были развиты от простого усреднения векторов признаков до сложных генеративных и дискриминативных моделей. На данный момент преобладают следующие методы создания моделей:

- для текстозависимых систем — динамическое искажение времени (Dynamic Time Warping; DTW) и скрытые марковские модели (Hidden

Markov Model; HMM);

- для текстонезависимых систем — векторное квантование (Vector Quantization; VQ), модели гауссовых смесей (Gaussian Mixture Model; GMM) и метод опорных векторов (Support Vector Machine; SVM).

Принятие решений осуществляют как с использованием единого классификатора, так и объединением решающих правил.

В части экспериментальной оценки пройден путь от лабораторных испытаний на небольших группах (5–10 человек) до создания представительных корпусов, отражающих реальные условия применения [12].

Литература

1. Будков В.Ю., Прищепа М.В., Ронжин А.Л., Марков К. Многоканальная система анализа речевой активности участников совещания // Третий междисциплинарный семинар «Анализ разговорной русской речи» АР^3. 2009. СПб. 2009. С. 57-62.
2. Рабинер Л.Р., Шафер Р.В. Цифровая обработка речевых сигналов / Пер. с англ. М. : Радио и связь, 1981. 496 с.
3. Садыхов Р.Х., Ракуш В.В. Модели гауссовых смесей для верификации диктора по произвольной речи // Доклады БГУИР. Минск, 2003. № 4. С. 95–103.
4. Сорокин В.Н., Цыплихин А.И. Верификация диктора по спектрально-временным параметрам речевого сигнала // Информационные процессы. 2010. Т. 10, № 2. С. 87-104.
5. Arya S., Mount D.M., Netanyahu N.S., Silverman R., Wu A. An optimal algorithm for approximate nearest neighbor searching in fixed dimensions // Journal of the ACM. 1998. V. 45, N 6. P. 891–923
6. Campbell J.P., Speaker Recognition: A Tutorial // Proceedings of the IEEE. 1997. V. 85, N 9. P. 1437-1462.
7. Martin A., Doddington G., Kamm T., Ordowski M., Przybocki M. The det curve in assessment of detection task performance // Proc. of Eurospeech. 1997. V. 4. P. 1895-1898.
8. Martin A., Przybocki M. The NIST 1999 Speaker Recognition Evaluation - An Overview // Digital Signal Processing. 2000. V. 10.
9. Navratil J., Klusacek D. On linear DETs // Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP-07). 2007.
10. Paredes R., Vidal E., Casacuberta F. Local features for speaker recognition // SPR 2004. International Workshop on Statistical Pattern Recognition. LNCS 3138 of Lecture Notes in Computer Science. 2004. P. 1087-1095.
11. Platt J.C. Fast Training of Support Vector Machines using Sequential Minimal Optimization // Advances in Kernel Methods / Ed. by B. Scholkopf, C.C. Burges, A.J. Smola. MIT Press, 1999. P. 185–208.
12. Reynolds D.A. An Overview of Automatic Speaker Recognition Technology // The International Conference on Acoustics, Speech, and Signal Processing ICASSP 02. 2002. P. 4072–4075.