

ОТКРЫТЫЕ СИСТЕМЫ МАССОВОГО ОБСЛУЖИВАНИЯ С РАСПОЗНАВАНИЕМ РЕЧИ

В.А. Жожикашвили, Р.В. Билик, В.А. Вертлиб, А.В. Жожикашвили,
Н.В. Петухова, М.П. Фархадов

Институт проблем управления им. В.А. Трапезникова, г. Москва

Обсуждена проблема доступа широких масс населения к информации. Введено понятие «открытые системы массового обслуживания» и рассмотрено применение технологии распознавания речи для организации голосового интерфейса информационных систем. Описана действующая система с телефонным доступом к базам данных.

ВВЕДЕНИЕ

Понятие *автоматизированные системы массового обслуживания (СМО)* впервые было предложено в трудах Института проблем управления (ИПУ) в 1965 г. и стало в дальнейшем общепринятым. Первой крупной СМО в СССР стала система «Сирена», предназначенная для бронирования мест и продажи билетов на самолеты Аэрофлота. В ней были реализованы наиболее передовые идеи вычислительной техники того времени, а именно, интерактивная удаленная база данных, сеть передачи данных с адаптивной маршрутизацией пакетов, экранные пульта с алфавитно-цифровой клавиатурой.

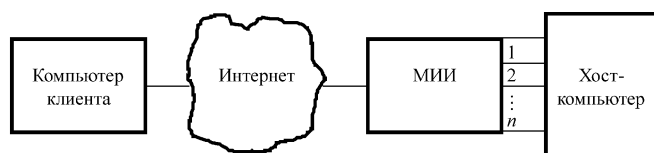
Вместе с тем «Сирена» по современным понятиям была закрытой системой в том смысле, что доступ к «своей» базе данных поддерживался только для узкого круга специально обученных операторов, в то время как доступа к чужим базам данных не было вовсе. С появлением Интернета «Сирена» стала эволюционировать в сторону большей открытости. Под открытостью системы понимается ее общедоступность.

Однако в настоящее время трансформация СМО из закрытой системы в открытую доступна только разработчикам или эксплуатационникам (только владельцам) СМО, которые в этом смысле являются монополистами, так как этот процесс требует серьезных усилий. Такое положение можно считать пережитком закрытого статуса СМО.

Несколько лет назад в ИПУ были начаты работы в области открытых систем обслуживания и разработаны принципы применения элементов естественного языка в человеко-машинном общении, позволяющие привлечь к системе большие массы клиентов, которые не проходили специального обучения [1, 2].

Структура СМО показана на рисунке. Клиент, не владеющий специальным языком запросов, обращается к системе на естественном языке. Перед хост-компьютером СМО, где размещена база данных, устанавливается компьютер, выполняющий функции модуля интеллектуального интерфейса (МИИ). Получив вызов клиента, МИИ выделяет из естественного языка смысл обращения (например, по ключевым словам) и подсказывает клиенту, что именно он должен передать в сторону хост-компьютера. При этом МИИ предлагает клиенту по возможности передать за один раз все необходимые для оформления сделки сведения. Получив их, МИИ формализует их в соответствии с требованиями хост-компьютера и располагает в необходимом порядке. Такой алгоритм позволяет достичь следующих результатов:

- обеспечивается доступ к услугам СМО массам необученных клиентов в среде Интернет, минуя агентов-операторов (телефонистов, диспетчеров или кассиров);
- не требуется никакого вмешательства в программы хост-компьютера;



Структура СМО с использованием элементов естественного языка

- минимизируется количество переключений «туда» и «обратно» в среде Интернет. В частном случае достигается одно обращение от клиента к базе данных хост-компьютера. Оно порождает n обращений от МИИ к базе данных, которые выполняются с высокой скоростью и не замедляют процесс заключения сделки.

Отметим еще две принципиальные особенности открытых СМО, получивших развитие в работах ИПУ:

- максимизация доходов от функционирования СМО (технология МАХ);
- применение технологии распознавания речи.

Технология МАХ начала применяться в странах Запада еще в 1990-х годах, но до настоящего времени практически не применяется в России. Ее компоненты:

- оптимальное управление ценами и тарифами,
- управление маршрутизацией товаров и услуг,
- сверхбронирование.

В наиболее развитой форме эти компоненты технологии МАХ получили воплощение в бизнесе авиаперевозок [3–5].

Однако, хотя технология МАХ и характерна для СМО нового поколения, она не вполне вписывается в понятие «открытые» системы и поэтому подробно не рассматривается в настоящей статье.

ПРИМЕНЕНИЕ ТЕХНОЛОГИИ РАСПОЗНАВАНИЯ РЕЧИ

Пожалуй, самым новым, оригинальным и перспективным стало применение в СМО технологии распознавания речи [6]. На сегодня она представляет собой наиболее естественный и многообещающий путь развития СМО, но одновременно это одна из самых сложных проблем в области человеко-машинного интерфейса. Распознавание речи – это ключ к максимальной открытости СМО, ибо оно доступно практически всем. В отличие от упомянутого алгоритма (см. рисунок), технология распознавания речи не требует наличия у клиента под рукой никакого компьютера, а МИИ дополняется программой распознавания речи. Технология распознавания речи, с одной стороны, работает в интересах клиентов, упрощая для них доступ к информации, с другой – в интересах фирм-провайдеров услуг и товаров, позволяя им сокращать

численность телефонистов, операторов и диспетчеров, занятых обслуживанием клиентов, и этим увеличивать доходность своего бизнеса.

Области применения открытых СМО с распознаванием речи настолько обширны и разнообразны, что подробно перечислять их нет необходимости. Укрупненно можно указать на следующие применения:

- всевозможные справочные системы;
- системы заказа и бронирования услуг и товаров;
- экспертные системы;
- управление механизмами, агрегатами и технологическими процессами;
- помощь людям с нарушениями зрения и двигательных функций рук.

Все известные алгоритмы распознавания речи работают на базе словарей, выделяя отдельные слова в речевом сигнале и преобразуя их в текст. Системы с распознаванием речи подразделяются на дикторозависимые и дикторонезависимые. Первые настраиваются на определенный голос и могут иметь словари объемом в несколько десятков тысяч слов, которые распознаются при слитном произношении. Вторые, напротив, настраиваются на различное произношение и способны содержать много меньшее количество слов, однако при большем разнообразии тематики. Они воспроизводят образцы речи сотен и даже тысяч носителей языка. Системы именно такого типа пригодны для открытых СМО.

Все компании пользуются примерно одинаковой базовой методикой, основанной на выполняемой в четыре этапа процедуре:

- *предпроцессор* выполняет предварительную обработку сигнала – автоматическую регулировку усиления, подавление эха и посторонних шумов, обнаружение начала и конца речи, обнаружение интонации конца фразы – и в результате на следующий этап передает сигнал высокого качества;
- *экстрактор* выполняет частотный анализ сигнала, акустическо-фонетический поток разбивается на короткие кадры (векторы) продолжительностью примерно 10 мс;
- *компаратор* сравнивает каждый кадр (вектор) с имеющимися фонетическими образцами, для каждой пары вычисляется функция правдоподобия;
- *интерпретатор* решает задачу динамического программирования с целью найти наилучшее разбиение полученного от компаратора алфавитного потока на слова и фразы, в зависимости от объема словаря и действующих синтаксических правил применяется та или иная стратегия поиска.



В результате распознавания речи выдается поток слов и предложений. Чтобы некоторое приложение могло воспользоваться этим продуктом, необходим прикладной программный интерфейс API.

Далее излагаются результаты, достигнутые в ИПУ РАН по тематике распознавания речи в СМО и существенно повышающие их открытость.

ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ ОТКРЫТЫХ СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ С РАСПОЗНАВАНИЕМ РЕЧИ

Главные отличительные особенности таких систем обслуживания, серьезно влияющие на процесс проектирования, заключаются в ориентации открытых систем на необученных пользователей с самым широким спектром вариантов их поведения и применение телефонов в качестве средства общения между ними и системой.

Процесс проектирования конкретного приложения, в котором предполагается автоматическое распознавание речи, состоит из этапов:

- изучения объекта и постановки задачи;
- проектирования приложения;
- реализации проекта;
- тестирования и настройки.

На первом этапе выявляются цели проекта, изучается технология, обследуется существующая техническая база.

Как правило, заказчик проекта имеет свою базу данных и операторов, которые принимают звонки клиентов, обращаются в базу данных за информацией и отвечают клиенту также по телефону. Во многих случаях такая система обслуживания уже бывает оборудована центрами обслуживания вызовов (Call Center, Contact Center) – программно-аппаратными комплексами, автоматизирующими процесс приема и обработки вызовов [7]. Call-центры включают в себя, как правило, интерактивную систему голосовых ответов IVR (Interactive Voice Response), позволяющую автоматизировать выдачу некоторых справок без обращения к оператору. Для этого система IVR зачитывает меню и предлагает выбрать нужный пункт путем нажатия соответствующей клавиши на клавиатуре тонального телефона. Абоненты с телефонными аппаратами без тонального набора не могут воспользоваться этой возможностью.

Отметим, что в нашей стране у большинства стационарных телефонных аппаратов пока нет опции тонального набора и, следовательно, они не годятся для самообслуживания в IVR-системах.

На этапе обследования объекта рассчитывается необходимое число портов, требуемая производительность обрабатывающего комплекса, срок окупаемости инвестиций, составляется бизнес-план. Если у заказчика уже есть IVR-система, то добав-

ление распознавания речи не потребует новых затрат на дооборудование существующих IVR-портов. Однако может потребоваться увеличение числа портов, тогда соответствующие затраты должны быть включены в расчет.

По оценкам западных источников стоимость обслуживания звонков с помощью IVR на порядок меньше, чем стоимость обслуживания звонков операторами. Практически то же соотношение имеет место для портов с возможностью распознавания речи. Поэтому даже при достаточно большой стоимости разработки проекта инвестиции окупаются за срок от шести месяцев до года. В наших условиях соотношение между стоимостью обслуживания звонков людьми и автоматом будет не таким впечатляющим по причине значительно меньших затрат на оплату работы операторов, однако и разработка проекта, и эксплуатационные расходы на поддержку работы оборудования тоже значительно меньше, чем на Западе, и по нашим оценкам инвестиции в проекты с распознаванием речи будут окупаться достаточно быстро.

На этом же этапе следует составить представление о клиентах: кто будет звонить в систему и как они будут реагировать на применение голосовой технологии. Это необходимо для последующего проектирования пользовательского интерфейса, хотя вполне возможно, что по мере дальнейшего знакомства с задачей представление о клиентах может трансформироваться. Следует также составить представление об имидже фирмы, для которой разрабатывается проект. Это может потребоваться, например, для решения вопроса какой нужен диктор для представления пользователю системы наиболее адекватным брэндом фирмы способом.

Результат первого этапа – согласованный обеими сторонами (заказчиком и разработчиком) План проекта.

Этап проектирования приложения, пожалуй, самый важный, поскольку здесь фактически разрабатывается технологический процесс. Хорошо спроектированная технология позволяет достичь вполне удовлетворительных результатов даже при ошибках в распознавании, которые неизбежны как вследствие недостаточного качества телефонной связи, так и по причине особенностей речи пользователей.

На этапе проектирования:

- пишутся сценарии обработки всех типов вызовов;
- определяется список грамматик;
- составляется спецификация промптов (текстов, произносимых системой) и намечается их содержание;
- определяется интерфейс с базой данных;
- выявляется необходимость модулей преобразования текста в речь и верификации голоса;

- составляются структурные схемы программ;
- записываются примеры аудиоклипов, демонстрирующие звучание голоса диктора.

На данном этапе необходима плотная работа с персоналом заказчика, владеющим всеми вопросами технологии, и исследование существующих диалогов между клиентами и операторами.

Результат этапа проектирования – спецификация пользовательского интерфейса, спецификация грамматик, структурные схемы программ обработки вызовов, примеры и спецификация промптов, спецификация интерфейса с базой данных, техническое задание на программирование.

На этапе реализации проекта намеченные задачи воплощаются в программные коды. Характерная особенность этого этапа заключается в создании специальной базы вызовов (речевого корпуса) для тестирования и настройки параметров.

На этапе реализации проекта:

- кодируется приложение;
- разрабатываются грамматики и создаются пользовательские словари;
- записываются все промпты;
- отлаживается интерфейс с базой данных;
- создается корпус для тестирования и настройки;
- настраивается модуль преобразования текста в речь, если предполагается его применение.

Этап реализации проекта заканчивается созданием системы, которая готова к тестированию.

Этап тестирования и настройки совершенно необходим для открытых систем с распознаванием речи, которые ориентированы именно на работу с весьма широким кругом пользователей. Прежде чем они начнут звонить в систему, разработчики должны сделать все возможное, чтобы пользователи приняли систему и оставались довольны ее работой. Для этого предназначен аппарат внутреннего тестирования и настройки параметров. Однако в практике эксплуатации таких систем широко применяются методы получения «обратной связи», суть которых состоит в накоплении записей о совершившихся диалогах и в специальной обработке этих записей, а также в получении прямой информации о работе системы от ряда клиентов, привлекаемых к таким опросам.

На этапе тестирования и настройки:

- осуществляется пилотное тестирование, выполняемое разработчиками, и настраиваются параметры;
- составляется программа ввода в опытную эксплуатацию;
- настраиваются параметры и дорабатываются сценарии и грамматики по результатам опытной эксплуатации;
- система вводится в эксплуатацию.

Особенности пользовательского интерфейса

Пользовательский интерфейс должен предоставлять пользователю как смысловую информацию, так и информацию о самом интерфейсе, а также принимать информацию от пользователя.

В рассматриваемых нами системах связь с пользователями осуществляется чаще всего по телефону, хотя в общем случае можно рассматривать в дополнение к таким клиентам еще и пользователей, обращающихся за тем же сервисом со своих компьютеров. В этом случае интерфейс комбинированный: голосовой и традиционный графический.

В голосовом пользовательском интерфейсе подача информации со стороны системы и реакция пользователя производятся в форме речи. Речь заменяет или дисплей, или кнопки, или индикаторы. Клиент пользуется телефоном – аппаратом, который изначально предназначался для взаимодействия с человеком на другом конце провода, поэтому он интуитивно ждет от компьютера такого же поведения, как от человека, да и сам часто разговаривает с компьютером, как с человеком.

Разумеется, идеалом была бы поддержка со стороны компьютера естественной речи человека. Работы в этом направлении ведутся в течение уже долгого времени, однако пока разработчики приложений имеют в своем распоряжении технологии с ограниченными возможностями, поэтому приближение диалога к естественному достигается в основном алгоритмическим путем.

Голосовой пользовательский интерфейс – относительно новая разновидность интерфейса «человек – машина», и для него пока нет установившихся правил, как для графического интерфейса [8], гарантирующих успех. Однако можно назвать несколько принципов, которые следует соблюдать и которые базируются на особенностях восприятия информации на слух, влиянии формы вопроса на ответ и на возможностях современных платформ распознавания речи.

Первый принцип, носящий самый общий характер и относящийся к проектированию любого интерфейса «человек-машина», состоит в том, что интерфейс должен быть ориентированным на человека и учитывать его возможности и слабости. Разработчик должен понимать трудности, с которыми будут сталкиваться люди при общении с создаваемым приложением. При разработке голосового интерфейса соблюдение этого принципа, может быть, даже важнее, чем в тех случаях, когда в цепочку общения включено зрение человека.

Другие принципы относятся уже непосредственно к голосовому интерфейсу:



- самый первый промпт должен содержать краткое объяснение, что и как делает данная система – клиент должен убедиться, что он позвонил в нужную ему систему, и понять, что сейчас он будет говорить с компьютером;
- при конструировании последующих промптов надо учитывать, что пользователь довольно быстро начинает понимать, как организован диалог, и усваивает терминологию; по мере обучения пользователя промпты-вопросы могут быть короче, а сообщаемая информация, наоборот длиннее;
- промпты должны быть однозначными, ясными и простыми по структуре;
- в промпте следует сообщать ровно столько информации, сколько необходимо пользователю для ее восприятия и однозначного ответа на задаваемый вопрос.

Помимо учета особенностей пользователя, разработчик интерфейса должен учитывать возможности машинной стороны. Что же предоставляют разработчику современные платформы распознавания речи?

Прежде всего отметим, что они обеспечивают достаточно высокий уровень распознавания: 95% и выше, что вполне достаточно для развертывания приложений. Неизбежные ошибки должны выявляться и корректироваться в ходе диалога. Для этого служит механизм подтверждения *Да/Нет*, однако включать его надо очень продуманно. Если задавать этот вопрос на каждом этапе получения данных от клиента, то диалог удлинится и клиент быстро начинает раздражаться. С другой стороны, клиент будет обеспокоен, если поймет, что он неправильно понял или что он дал неправильный ответ на заданный вопрос, и будет торопиться исправить ошибку вместо того, чтобы заняться ответом на следующий вопрос. Однако это уже вопросы, относящиеся к задачам управления диалогом.

Далее, неплохим инструментом для сокращения продолжительности диалога служит программно-аппаратная поддержка функции *Barge-In*, когда клиент может отвечать на задаваемый компьютером вопрос, не дожидаясь окончания машинного промпта. О наличии такой возможности приложение должно сообщать клиенту в начале своей работы. Пользователь-новичок вряд ли воспользуется этой опцией. Но как только он поймет, что это такое, он будет ее использовать, тем самым значительно сокращая время диалога. Клиенты обучаются довольно быстро и, тем более, таким вещам, которые позволяют им сэкономить время. Функция *Barge-In* может быть реализована в самом хосте, но гораздо предпочтительнее голосовые платы с поддержкой функции *Barge-In*, чтобы освободить хост-компьютер от этой работы. Функция *Barge-In*

работает вместе с функциями определения начала и конца речи и эхоподавления. Для ее эффективной работы необходимо учитывать некоторые ограничения на грамматику, применяемую в сочетании с этой функцией. Грамматика должна оперировать по возможности простыми словами и фразами и не быть иерархической.

Еще один инструмент для конструирования интерфейса представляют собой открытые грамматики, предлагаемые разработчиками ряда платформ распознавания речи. Открытые грамматики дают пользователю значительно большую свободу высказывания в ответ на вопрос системы. Для системы важно, чтобы в ответе присутствовали ожидаемые ключевые слова, а все другие слова игнорируются. Открытые грамматики позволяют клиенту говорить более естественно, что очень важно для интерфейса открытых систем, ориентированных на массового пользователя. Однако такие грамматики требуют настройки и тренинга, что весьма трудоемко.

Конструкторы грамматик предлагают и другие механизмы для приема более натуральной речи от клиента. Они состоят в том, например, что в грамматику включается несколько предвидимых вариантов ответов, допускаются разные варианты порядка слов в ответах, заранее устанавливается более низкий порог распознавания для необязательных слов.

Можно назвать и другие средства систем распознавания, которые разработчик может применить при конструировании интерфейса, но тут важно отметить одно важное обстоятельство, а именно, что базовое матобеспечение платформ распознавания речи постоянно развивается и что эти усовершенствования в значительной своей части направлены именно на создание более натурального голосового интерфейса «человек – машина».

Управление диалогом в открытых системах

Цель управления диалогом – обеспечение успешного обслуживания вызова за кратчайшее время.

Управление диалогом необходимо даже в простейших системах, поскольку надо предвидеть возможные ошибки в распознавании и различные варианты поведения пользователя. В сложных системах с продолжительным диалогом и с выводом большого объема информации клиенту количество точек, где необходимо управлять процессом, бывает значительным.

Задача минимизации времени обслуживания вызова должна решаться еще на этапе планирования сценария. Для этой цели применяются вероятностные модели, которые описывают состояния

процесса, характеризующиеся временами пребывания в них и вероятностями переходов между состояниями.

Такой подход позволяет выбрать наиболее оптимальный сценарий, но это будет планирование на макроуровне. Для приложений со сложным и продолжительным диалогом со многими разветвлениями этих действий недостаточно.

Для таких систем помимо решения задачи оптимизации на этапе планирования сценария закладывается также возможность динамического управления диалогом в ходе его выполнения. Управление реализуется обычно в точках получения результата от модуля распознавания и в точках подготовки к выводу информации, полученной из базы данных.

Модуль распознавания может выдать результат распознавания и уверенность (вероятность), с которой этот результат получен, либо в случае неуспеха ошибку распознавания и ее тип. Если используется опция *N-best*, то выдается не один результат, самый вероятный, а *N*, каждый со своей гипотезой и ее вероятностью.

Если уверенность в распознавании основной (первой) гипотезы больше некоторого заданного значения (не путать с порогом уверенности, который разграничивает успех и неудачу в распознавании), то модуль управления продвигает диалог к следующему шагу. В противном же случае предпринимаются некоторые действия, нацеленные на то, чтобы отметить, что распознавание в этой точке было выполнено с недостаточно высокой уверенностью. Эти данные могут быть использованы далее, а пока управление может быть возвращено в ту же точку, после чего диалог также продвигается к следующему шагу. Если используется опция *N-best*, то управление может быть более сложным. Например, при низкой уверенности основной гипотезы и достаточно высокой уверенности второй гипотезы необходимо проанализировать, на каком слове какой грамматики это произошло. Дело в том, что часто не удается избежать включения в грамматику похожих по звучанию слов, например, *июнь* и *июль* в грамматике месяцев, или очень коротких слов со свистящими звуками, которые распознаются как правило хуже. При разработке стратегии управления можно учитывать такие факторы и если обнаруживается низкая уверенность распознавания основной гипотезы, а также есть и вторая гипотеза с относительно высокой вероятностью, то диалог может быть направлен на веточку с промптом типа «*Вы сказали июнь, да или нет?*».

Другой тип управления можно назвать ситуационным. Суть его состоит в том, что в расчет принимаются такие факторы, как история диалога,

наличие ошибок на предыдущих фазах, объем предназначенной для вывода информации. К примеру, если произошла ошибка в распознавании, то необходимо проанализировать, первая ли она на этом шаге диалога или нет, и в зависимости от этого направить клиенту промпт того или иного содержания. Или если объем данных, полученных из базы, велик, то в промпте надо предупредить клиента об этом и предложить выбрать способ передачи ему этой информации.

Существуют и другие виды управления диалогом, при умелом применении которых можно значительно повысить удовлетворенность клиента работой системы.

Пути улучшения качества распознавания

Базовые *soft-средства*, используемые в системах распознавания, обычно позволяют выполнить настройку системы с тем, чтобы повысить надежность и достоверность распознавания. К сожалению, настройка системы не приводит к коренному улучшению распознавания, а позволяет лишь улучшить поведение системы в отношении одного из показателей качества за счет ухудшения другого.

Один из основных настраиваемых параметров — порог уверенности распознавания. Он представляет собой такое значение вероятности некоторой гипотезы, которое является пороговым для принятия решения о том, принять данную гипотезу или отвергнуть ее.

При выборе порога уверенности во внимание принимаются следующие соображения:

- если установить слишком высокий порог уверенности, то будут частыми случаи, когда слово или фраза распознаны верно, но отвергнуты модулем распознавания;
- если порог уверенности занижен, то будет много случаев, когда сказанная клиентом фраза распознана неверно, но модуль распознавания выдал свою гипотезу в качестве верной.

Рекомендуемый разработчиками базовых систем порог уверенности равен 45 (при шкале от 1 до 100). Считается, что при таком пороге обеспечивается достаточно уверенное распознавание с небольшим количеством ошибочного распознавания и отказов в распознавании.

Порог распознавания может быть задан как по отношению ко всему приложению, так и по отношению к конкретной грамматике, группе слов и к отдельному слову.

Другой важный фактор, влияющий на качество распознавания, — сама грамматика. Понятие грамматики в системах распознавания речи отличается от принятого в лингвистике. Здесь под граммати-



кой понимается последовательность слов, которые модуль распознавания способен понять на данной фазе диалога. Грамматики могут быть простыми, содержащими перечни слов или групп слов, соединенных логическими союзами И или ИЛИ, но могут быть и сложными, включающими в себя другие грамматики или рекурсию.

Известно, что чем проще грамматика, тем лучше будет распознавание, однако не всегда это возможно обеспечить, хотя всегда следует к этому стремиться.

Каждой грамматике соответствует свой словарь, содержащий транскрипцию произношения всех слов грамматики. Этот словарь создается компилятором на основе базового словаря, а транскрипцию слов, отсутствующих в нем, разработчик должен написать сам. Хорошо известно, что следует избегать использования в одной грамматике слов, имеющих схожее произношение, но этого тоже не всегда удается достичь. Разработчик должен очень внимательно отнестись к содержимому словарей, поскольку часто некорректное описание произношения приводит к серьезным ошибкам в распознавании.

Базовое матобеспечение содержит обычно несколько наиболее распространенных грамматик, которые рекомендуется использовать в приложениях, а также базовый словарь достаточно большого объема.

СОСТОЯНИЕ ДЕЛ В МИРЕ

В настоящее время на Западе уже сотни компаний внедрили у себя системы распознавания речи, чтобы автоматизировать обслуживание различного рода вызовов. Одними из первых применили у себя эту технологию авиакомпании, среди которых «American Airlines», «United Airlines» и др. Широко применяется распознавание речи банками, страховыми компаниями, правительственными учреждениями, операторами связи. Инвестиции в технологии распознавания речи составили в 2002 г. 695 млн. дол. США и ожидается, что они возрастут к концу 2004 г. до 1,4 млрд. дол. [9].

В нашей стране работы по распознаванию речи велись в течение многих лет, и достигнутые результаты были на очень высоком теоретическом уровне. Однако систем, пригодных для практического применения, создано не было. В настоящее время в России нет ни одной действующей системы массового обслуживания населения с распознаванием русской речи.

СИСТЕМА С ГОЛОСОВЫМ ИНТЕРФЕЙСОМ

В Институте проблем управления РАН создана система доступа к информации СМО «Сирена», которая демонстрирует возможности применения технологии распознавания речи в открытых системах массового обслуживания [10]. Любой человек с любого телефона может набрать нужный номер. Компьютер снимет трубку, ответит ему и задаст несколько вопросов, чтобы узнать, какая информация нужна позвонившему. Ответы распознаются, и на основании полученной информации компьютер формирует запрос в «Сирену». Затем ответ, полученный из «Сирены», озвучивается и сообщается клиенту. Система может распознавать числа, названия улиц, номера телефонов, цифровые коды.

В процессе разработки системы были созданы грамматики, фонетические словари, записаны звуковые файлы, построен диалог. Пример диалога приводится далее.

Система построена на компьютере Pentium 4, дополненном платой Dialogic, к которой подключены две телефонные линии. Операционная система – Windows NT. Применены платформы распознавания речи компаний «Nuance Communication» и «Philips».

Пример диалога с системой «Сирена»

С и с т е м а приветствует клиента и предлагает выбрать одну из тем: месяцы, числа, расписание рейсов.

К л и е н т. Расписание рейсов.

С и с т е м а. Назовите город отправления.

К л и е н т называет город.

С и с т е м а (для проверки произносит название города). Назовите город назначения.

К л и е н т называет город.

С и с т е м а (повторяет город). Назовите месяц вылета.

К л и е н т называет любой месяц.

С и с т е м а (повторяет месяц). Назовите число месяца.

К л и е н т называет любое число в пределах от 1 до 31 (можно говорить «второе» или «второго»).

С и с т е м а. Ваш запрос содержит: (перечисляет параметры запроса). Это правильно или нет?

К л и е н т говорит «нет» или «неправильно» или «нет, неправильно», если была ошибка, или «правильно», «да, правильно», если все верно.

В случае, если клиент говорит «нет»:

С и с т е м а. Что вы хотите изменить: город отправления, город назначения, месяц вылета или число месяца?

К л и е н т (например). Число месяца.

И т. д.

Этот диалог используется далее для обращения в реальную базу данных действующей системы «Сирена». После получения ответа из «Сирены» система озвучивает его, называя количество выполняемых в данный день рейсов, номера рейсов, времена вылета и прилета, базовый тариф и другую информацию.

Эксперименты показали, что система вполне удовлетворительно распознает речь клиентов, а также демонстрирует высокую реактивность.

Из приведенного диалога ясно, что в этой разработке использовались грамматики, обеспечивающие распознавание часто употребляемых слов: чисел, названий месяцев, городов. Поскольку создан ряд и других общеупотребительных грамматик (номера телефонов, цифровые коды, улицы Москвы и др.), можно утверждать, что данная система достаточно универсальна и может найти применение в различных областях.

ЗАКЛЮЧЕНИЕ

Информационные системы давно уже вошли в повседневную жизнь человека. Информации становится все больше, и все более актуальной становится проблема доступа к ней. Современное состояние информационных технологий позволяет находить новые подходы к решению проблемы доступа. Одно из таких направлений – применение технологии распознавания речи, весьма перспективной и способствующей эволюции информационных систем в сторону открытых систем массового обслуживания.

ЛИТЕРАТУРА

1. *Жожикашвили В.А., Жожикашвили А.В., Фархадов М.П.* Принципы построения системы общения с использованием естественного языка в больших автоматизированных системах массового обслуживания нового поколения // Массовое обслуживание. Поток, системы, сети: Сб. матер. 14 Белорус. зимней школы-семинара по теории массового обслуживания (ВВWQT-98). – Минск, 1998. – С. 190–194.
2. *Жожикашвили В.А., Жожикашвили А.В., Петухова Н.В.* и др. Построение структуры человеко-машинного интерфейса для автоматизированных систем массового обслуживания // Автоматизация и современные технологии. – 2000. – № 3.
3. *Жожикашвили В.А., Билик Р.В., Петухова Н.В., Ребортович Б.И.* Управление доходами в автоматизированных системах массового обслуживания. // Тр. Ин-та пробл. упр. РАН. – 1999. – Т. IV. – С. 27–35.
4. *Зутлер И.А., Фархадов М.П.* Стратегия управления сверхбронированием и сверхпродажами // Тр. Ин-та пробл. упр. РАН. – 2000. – Т. IX. – С. 54–58.
5. *Зутлер И.А.* Управление ограниченным ресурсом на конечном промежутке времени в автоматизированных системах массового обслуживания // Автоматика и телемеханика. – 2002. – № 2. – С. 179–184.
6. *Жожикашвили В.А., Жожикашвили А.В., Петухова Н.В., Фархадов М.П.* Применение распознавания речи в автоматизированных системах массового обслуживания // Автоматизация и современные технологии. – 2003 (в печати).
7. *Росляков А.В., Самсонов М.Ю., Шибяева И.В.* Центры обслуживания вызовов (Call Centre). – М.: Эко-Трендз, 2002.
8. *Raskin J.* The Human Interface: New Directions for Designing Interactive Systems. – Addison-Wesley: Pearson Education, Inc., 2000.
9. <http://www.datamonitor.com>
10. <http://www.ipu.ru/labs/lab17/frame17.htm>

☎ (095) 334-87-10



Читайте в следующем номере нашего журнала

❖ **Эпштейн В.Л.** Предвидимое будущее научных журналов (обзор)

❖ **Бабаян Р.Р., Морозов В.П.** Надо ли возрождать аналоговую технику?

❖ **Нижегородцев Р.М.** Логическое моделирование экономической динамики. Ч. I.

❖ **Лебедев В.Н.** Управление синхронизацией распределенных разнородных баз данных на основе организации трехэлементных очередей

❖ **Кульба В.В., Темкин В.М., Рывкин Д.Б.** Сбор и обработка данных в системах производственной экологической безопасности предприятий нефтегазовой промышленности