(19) Евразийское патентное ведомство

(21) 201891077 (13) A1

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ЕВРАЗИЙСКОЙ ЗАЯВКЕ

(43) Дата публикации заявки
2019.12.30

(22) Дата подачи заявки
2018.05.31

(51) Int. Cl. ***G10L 15/22*** (2006.01)
G06F 17/20 (2006.01)

(54) СПОСОБ И СИСТЕМА ДЛЯ ВЫСТРАИВАНИЯ ДИАЛОГА С ПОЛЬЗОВАТЕЛЕМ В УДОБНОМ ДЛЯ ПОЛЬЗОВАТЕЛЯ КАНАЛЕ

(31) **2018120199**
(32) **2018.05.31**
(33) **RU**

(71) Заявитель:
ПУБЛИЧНОЕ АКЦИОНЕРНОЕ ОБЩЕСТВО "СБЕРБАНК РОССИИ" (ПАО СБЕРБАНК) (RU)

(72) Изобретатель:
Кузнецов Никита Александрович, Кирьянов Денис Павлович, Чернопятов Андрей Сергеевич, Доманская Кристина Сергеевна (RU)

(74) Представитель:
Герасин Б.В. (RU)

(57) Настоящее техническое решение, в общем, относится к автоматизированным информационным системам, а более конкретно к технологии для интеллектуального генерирования диалоговых сообщений для выстраивания диалога с пользователем. Способ для выстраивания диалога с пользователем в удобном для пользователя канале включает получение пользовательского ввода в графическом интерфейсе пользователя, дальнейшую предобработку полученного текста посредством использования синтаксического парсера, а также лемматизацию, исправление орфографических ошибок, после чего классификацию пользовательского ввода для отнесения к тому или иному диалоговому сценарию, и в итоге формирование ответа для пользователя. Технический результат - увеличение скорости обслуживания пользователя.

201891077 A1

201891077 A1

G10L15/22

СПОСОБ И СИСТЕМА ДЛЯ ВЫСТРАИВАНИЯ ДИАЛОГА С ПОЛЬЗОВАТЕЛЕМ В УДОБНОМ ДЛЯ ПОЛЬЗОВАТЕЛЯ КАНАЛЕ

ОБЛАСТЬ ТЕХНИКИ

[001] Настоящее техническое решение, в общем, относится к автоматизированным информационным системам а, более конкретно, к технологии для интеллектуального и проактивного генерирования диалоговых сообщений для выстраивания диалога с пользователем.

УРОВЕНЬ ТЕХНИКИ

[002] Технические решения, описанные в этом разделе, могут осуществляться, но необязательно представляют собой способы и системы, которые ранее рассматривались или осуществлялись. Следовательно, если только не указано иное, не следует исходить из того, что любое из технических решений, описанных в этом разделе, квалифицируется как предшествующий уровень техники только в силу их включения в этот раздел.

[003] Чат-бот как информационная автоматизированная система представляет собой агент на основе компьютера, имеющий адаптированный для человека графический интерфейс для доступа и управления информацией. Традиционно, чат-бот может взаимодействовать с пользователями на естественном языке для симуляции интеллектуального общения и предоставления персональной помощи пользователям. Например, пользователи могут задавать в чат-боте такие вопросы как, например, «Где находится ближайший отель?» или «Какая сейчас погода?», и получать соответствующие ответы. Пользователи также могут просить в чат-боте выполнить определенные функции, включая, например, создание электронных писем, выполнение телефонных звонков, поиск информации, получение данных, переадресацию пользовательских запросов, направление пользователя, обеспечение заметок и напоминаний и так далее. Чат-боты и системы персональных цифровых помощников широко применяются и оказывают огромную помощь для пользователей компьютеров и являются особенно полезными для владельцев портативных электронных устройств, например смартфонов, планшетных компьютеров, игровых консолей и так далее.

[004] Термин «чат-бот» также может быть известен и использоваться как «разговорная диалоговая система», «диалоговая система», «коммуникационный агент», «робот-собеседник», «бот–собеседник», «чат агент», «цифровой персональный помощник/агент»,

«автоматизированный онлайн помощник» и так далее. Все эти термины подпадают под объем настоящего описания технического решения.

[005] По существу, пользователи чат-ботов могут задавать большое количество различных вопросов и запрашивать большой диапазон информации, относящейся к мировым и местным новостям, погоде, контенту электронной почты, календарным встречам, запланированным событиям и любому другому доступному для поиска контенту. Чат-бот может быть полезным не только для доступа к определенной информации, но также для генерирования контента, планирования событий, написания электронных писем, навигации, и гораздо большего. С другой стороны, часто для пользователей трудно понять, какой тип информации может запрашиваться через чат-бот в конкретный момент времени. Например, начинающие пользователи могут с трудом понимать или знать принципы работы чат-бота или его конкретные функциональные возможности. Пользователи могут недопонимать, что несколько особенностей приложения чат-бота могут очень часто применяться для решения их повседневных задач и потребностей. Следовательно, по-прежнему существует необходимость развития чат-ботов и, в частности, существует необходимость усовершенствования интерфейса взаимодействия человека с чат-ботом.

[006] В финансово-кредитных учреждениях нагрузка на колл-центры постоянно растет, причем количество звонков и диалогов с операторами исчисляется миллионами в год. Большая часть вопросов является прогнозируемой и не требует уникального ответа, причем может закрываться в режиме реального времени автоматически.

СУЩНОСТЬ ТЕХНИЧЕСКОГО РЕШЕНИЯ

[007] Данное техническое решение направлено на устранение недостатков, присущих существующим решениям, известным из уровня техники.

[008] Технической проблемой (или технической задачей) в данном техническом решении является формирование диалога с пользователем, позволяющего точно ответить на возникшие вопросы пользователя.

[009] Техническим результатом, проявляющимся при решении вышеуказанной задачи, является увеличение скорости обслуживания пользователя.

[0010] Дополнительным техническим результатом является снижение вычислительной нагрузки на информационную систему колл-центра за счет снижения количества звонков пользователей и количества диалогов в службе поддержки.

[0011] Указанный технический результат достигается благодаря реализации способа для выстраивания диалога с пользователем в удобном для пользователя канале, в котором

получают посредством процессора, функционально соединенного с базой данных, данные пользовательского ввода; осуществляют предобработку пользовательского ввода посредством его разбиения на предложения и слова, причем исправляют орфографические ошибки пользователя посредством использования модуля исправления опечаток; выполняют лемматизацию каждого слова из пользовательского ввода; формируют структуру зависимостей слов друг от друга в пользовательском вводе посредством использования синтаксического парсера; формируют векторную модель слов пользовательского ввода; классифицируют посредством модуля ведения диалога с пользователем по меньшей мере часть пользовательского ввода для формирования ответа; предоставляют посредством процессора по меньшей мере один ответ на распознанный пользовательский ввод данных.

[0012] В некоторых вариантах реализации при осуществлении предобработки пользовательского ввода дополнительно исправляют опечатки в пользовательском вводе посредством подмодуля исправления опечаток.

[0013] В некоторых вариантах реализации при осуществлении предобработки пользовательского ввода дополнительно выполняют токенизацию числительных из пользовательского ввода.

[0014] В некоторых вариантах реализации синтаксический парсер текста для анализа пользовательского текста использует рекуррентные нейронные сети.

[0015] В некоторых вариантах реализации при классификации пользовательского ввода модуль ведения диалога использует алгоритм заполнения слотов.

[0016] В некоторых вариантах реализации при классификации посредством модуля ведения диалога с пользователем пользовательского ввода определяют линейные сценарии диалога или нелинейные сценарии диалога.

[0017] В некоторых вариантах реализации при формировании векторной модели слов пользовательского ввода используют модель CBOW и/или Skip-gram.

[0018] В некоторых вариантах реализации при формировании ответа для пользователя для отображения в графическом интерфейсе пользователя определяют класс, к которому относится пользовательский ввод.

[0019] В некоторых вариантах реализации при формировании ответа для пользователя для отображения в графическом интерфейсе пользователя в случае отсутствии ответа в диалоге, переключают его на оператора.

КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

[0020] Признаки и преимущества настоящего изобретения станут очевидными из приводимого ниже подробного описания изобретения и прилагаемых чертежей, на которых:

[0021] На Фиг. 1 показан пример реализации системы для выстраивания диалога с пользователем в удобном для пользователя канале;

[0022] На Фиг. 2 показан пример реализации модуля предобработки текста;

[0023] На Фиг. 3 показан пример реализации перехода фразы из одной в другую согласно подходу расстояния движения слов.

[0024] На Фиг. 4 показан пример реализации способа для выстраивания диалога с пользователем в удобном для пользователя канале.

[0025] На Фиг. 5 показан пример реализации подхода word2vec, который позволяет оценивать семантическую близость слов.

[0026] На Фиг. 6 пример реализации системы для выстраивания диалога с пользователем в удобном для пользователя канале, реализованной посредством набора вычислительных компонент.

ПОДРОБНОЕ ОПИСАНИЕ ИЗОБРЕТЕНИЯ

[0027] Данное техническое решение может быть реализовано на компьютере, в виде автоматизированной информационной системы (АИС) или машиночитаемого носителя, содержащего инструкции для выполнения вышеупомянутого способа.

[0028] Техническое решение может быть реализовано в виде распределенной компьютерной системы.

[0029] В данном решении под системой подразумевается компьютерная система, ЭВМ (электронно-вычислительная машина), ЧПУ (числовое программное управление), ПЛК (программируемый логический контроллер), компьютеризированные системы управления и любые другие устройства, способные выполнять заданную, чётко определённую последовательность вычислительных операций (действий, инструкций).

[0030] Под устройством обработки команд подразумевается электронный блок либо интегральная схема (микропроцессор), исполняющая машинные инструкции (программы).

[0031] Устройство обработки команд считывает и выполняет машинные инструкции (программы) с одного или более устройства хранения данных. В роли устройства хранения данных могут выступать, но, не ограничиваясь, жесткие диски (HDD), флеш-память, ПЗУ (постоянное запоминающее устройство), твердотельные накопители (SSD), оптические приводы.

[0032] Программа - последовательность инструкций, предназначенных для исполнения устройством управления вычислительной машины или устройством обработки команд.

[0033] Ниже будут описаны термины и понятия, необходимые для реализации настоящего технического решения.

[0034] Виртуальный собеседник, программа-собеседник, чат-бот (англ. chatbot) — компьютерная программа, имитирующая речевое поведение человека при общении с одним или несколькими собеседниками.

[0035] Лемматизация — процесс приведения словоформы к лемме — её нормальной (словарной) форме. Например, «кошками» после лемматизации преобразовывается в «кошка».

[0036] База данных (БД) - совокупность данных, организованных в соответствии с концептуальной структурой, описывающей характеристики этих данных и взаимоотношения между ними, причем такое собрание данных, которое поддерживает одну или более областей применения (ISO/IEC 2382:2015, 2121423 «database»).

[0037] Как показано на Фиг. 1, система **100** для выстраивания диалога с пользователем в удобном для пользователя канале передачи данных может содержать описанные ниже модули, между которыми осуществляется обмен данными.

[0038] Предварительно в модуль **120** предобработки текста поступает запрос на осуществление той или иной услуги от пользователя (или некоторая его модификация), который его формирует посредством своего графического интерфейса **110** пользователя, например, в мобильном приложении на мобильном устройстве связи.

[0039] Услуга, которую запрашивает пользователь, может включать, например, получение кредита, выпуск пластиковых карт, депозиты, лизинг, обслуживание расчетного счета, операции с иностранной валютой и т.д.

[0040] На данном шаге **401**, как это показано на Фиг. 4, фактически получают пользовательский ввод данных, который может представлять собой строку или их набор, причем символьную. Пользовательский ввод данных может включать как один вопрос, так и их набор. Данные пользовательского ввода сохраняются в базу данных **150**. Строка может быть, не ограничиваясь, например, следующего вида: «Привет, чат-бот! Переведи Денису Иванову с моего счета 100 рублей на его телефон».

[0041] Базовый модуль **120** предобработки текста, отвечает за снижение многообразия возможных текстов сообщений для того, чтобы упростить работу следующим модулям системы **100**. Данный модуль предназначен для разбиения входящего пользовательского ввода на предложения и слова, а также их морфологического разбора, синтаксического разбора и семантической типизации токенов.

[0042] Этап токенизации, предполагает выделение базовых элементов текста (токенов), ограниченных с двух сторон разделительными символами, пробелами или знаками пунктуации. Элементами здесь выступают слова, числа, даты, сокращения, аббревиатуры, составные предлоги и т.д. Токенизация позволяет выделить дискретные единицы текста, являющиеся основой для дальнейшей работы на этапах морфологического и синтаксического анализа. В результате токенизации каждому элементу присваивается соответствующий тип: слово, число, дата, адрес и т.д.

[0043] Цель предобработки текста - подготовить его для качественной классификации запросов. Для этого запрос проходит много этапов предобработки **402** и обогащения различной информацией, среди которых основными этапами являются:

[0044] исправление опечаток;

[0045] лемматизация со снятием морфологической омонимии;

[0046] обогащение результатами разбора синтаксического парсера и отдельного выделения набора троек объект+субъект+действие (SAO);

[0047] перевод числительных из текста в цифровое представление ("тыща триста" заменяется на 1300), выделение ФИО, времени и т.д.

[0048] Модуль **120** предобработки текста включает, не ограничиваясь, как показано на Фиг. 2, синтаксический парсер **210** текста, подмодуль **220** исправления опечаток, морфологический анализатор **230** русского языка с поддержкой снятия морфологической неоднозначности, подмодуль **240** замены синонимов и коллокаций, подмодуль **250** замены числительных, денежных сумм и номеров телефонов.

[0049] Подмодуль **220** исправления опечаток используется для исправления орфографических ошибок пользователей, что помогает правильно классифицировать запрос.

[0050] Принцип работы данного подмодуля **220** следующий:

[0051] разбивают предложение на слова;

[0052] для каждого слова:

a. если длина слова больше 2 - возвращают его;
b. если есть типовое исправление ошибки по словарю - возвращают типовое исправление;
c. если слово состоит не только из букв - возвращают его;
d. если слово знакомо частотному словарю - возвращают его
e. если слово не знакомо, возвращают наиболее вероятное исправление:

i. генерируют все варианты исправления слова с одним изменением: удаление, добавление, замена одиночных символов или замена местами двух соседних в слове;
ii. отфильтровывают только знакомые частотному словарю, выделяют их долю (p1);
iii. если кандидат 1 - возвращают его;
iv. выделяют вероятность конкретной ошибки (p2) (например, вероятность ошибки е/и выше, чем е/ь);
v. выделяют вероятность сочетания подобранных вариантов исправления с предыдущим словом (p3);
vi. вычисляют совокупную вероятность каждого кандидата p=p1^λ1 * p2^λ2 * p3^λ3, где λ1, λ2 и λ3 - настраиваемые параметры, и возвращают наилучший.

[0053] Морфологический анализатор **230** русского языка с поддержкой снятия морфологической неоднозначности представляет собой набор алгоритмов, которые занимается сопоставлением отдельных слов и словоформ в словаре (лексиконе, если быть точным) и выяснением грамматических характеристик слов. Разметка исходного текста грамматической информацией очень облегчает составление правил далее при работе синтаксического парсера **210** текста. Работа морфологического анализатора **230** финализируется установлением морфологических признаков слов текста. Эта задача может рассматриваться как задача разметки или тегирования текста – установления тегов (морфологических признаков). Набор устанавливаемых признаков при этом непосредственно зависит от языка. Морфологический анализатор **230** в некоторых вариантах реализации может осуществлять морфологический анализ, который может быть словарным (со словарем основ и окончаний или словарем словоформ) или бессловарным (только со словарем окончаний; словарь окончаний может быть встроен в алгоритм морфологического анализа **230**). Бессловарный метод используется только для определения переменной морфологической информации (не всегда однозначно), а словарный — во всех остальных случаях. Алгоритмы, модель формообразования работы морфологического анализатора **230**, который осуществляет генерацию и определение словоформ и примеры для различных естественных языков известны из источника информации [3].

[0054] Подмодуль **240** замены синонимов и коллокаций выполнен с возможностью определения синонимов и коллокаций в пользовательском вводе и имеющейся базе данных **150** на основании статистических мер (MI, t-score, log-likelihood), которые чаще

всего используются при определении степени близости между компонентами словосочетаний в корпусе. Мера MI позволяет выделить устойчивые словосочетания, имена собственные, а также низкочастотные специальные термины. Слова, у которых мера MI-score принимает наибольшую величину, менее частотны и обладают ограниченной сочетаемостью. Мера t-score также учитывает частоту совместной встречаемости ключевого слова и коллоката, отвечая на вопрос, насколько не случайной является сила ассоциаций (связанности) между коллокатами. Слова с наибольшим значением t-score оказываются частотными и могут сочетаться с множеством единиц. В некоторых вариантах реализации подмодуль **240** замены синонимов и коллокаций может использовать каждую приведенную меру или несколько в совокупности.

[0055] Результатом работы синтаксического парсера **210** текста является структура зависимостей слов друг от друга в пользовательском вводе в этом тексте (отдельных его предложениях). Например, из фразы «переведи маме сто рублей» система **100** принимает решение, что мама — это косвенный объект (адресат).

[0056] Синтаксический парсер **210** текста выделяет синтаксическую структуру предложения, которая представляет собой дерево зависимостей, в узлах которого стоят слова предложения, а ветви помечены именами синтаксических отношений.

[0057] Работа синтаксического парсера **210** текста уже является распознаванием пользовательского ввода **403**, как это показано на Фиг. 4.

[0058] В некоторых вариантах реализации синтаксический парсер **210** текста для анализа пользовательского ввода может использовать для своей работы рекуррентные нейронные сети (Recurrent Neural Network, RNN) для обучения синтаксического парсера **210** текста. Рекуррентные нейронные сети - это класс моделей машинного обучения, основанный на использовании предыдущих состояний сети для вычисления текущего. Каждый символ в исходном тексте, отдельные слова, знаки препинания и даже целые фразы — все это является атомарным элементом входной последовательности для нейронной сети. В некоторых вариантах реализации могут использоваться управляемые рекуррентные нейроны (gated recurrent units, GRU). Фильтр обновления определяет, сколько информации останется от прошлого состояния и сколько будет взято из предыдущего слоя. Фильтр сброса работает примерно как забывающий фильтр.

[0059] Для обучения нейронных сетей в качестве обучающей выборки используются корпусы, а именно трибанки. В лингвистике корпус — подобранная и обработанная по определённым правилам совокупность текстов, используемых в качестве базы данных для исследования языка. Они используются для статистического анализа и проверки статистических гипотез, подтверждения лингвистических правил в данном языке. Трибанк

— это коллекция разобранных предложений (то есть графов разбора), подготовленных вручную или автоматически заранее. Трибанки по классификации делятся на phrase-structure treebanks и dependency treebanks. В данном техническом решении могут использоваться, не ограничиваясь, следующие трибанки или корпусы для русского языка: SynTagRus (1,107 тыс. токенов), PUD (19 тыс. токенов), GSD (99 тыс. токенов), Taiga (20 тыс. токенов), Dependency Treebanks и т.д.

[0060] Далее применяют подход Transition-based dependency parsing, широко известный из уровня техники. Этот подход заключается в попытке предсказать последовательность действий (переходов) от некоторой изначальной конфигурации фразы или запроса пользователя к конечной, в результате выполнения которых будет получено искомое дерево разбора, что позволяет получить достаточно высокую точность и достичь довольно высоких скоростей при обработке текста.

[0061] Arc-standard system – один из наиболее популярных подходов реализации transition-based системы. Система описывается конфигурацией, состоящей из трех частей: c = (s, b, A),

[0062] где: s – стек данных;

[0063] b – буфер данных;

[0064] A – множество зависимостей.

[0065] Изначально конфигурация для последовательности символов $w_1, \ldots, w_n$ следующая перед обработкой:

[0066] s = [ROOT] – в стеке один служебный символ;

[0067] b = $[w_1, \ldots, w_n]$ – в буфере вся последовательность символов;

[0068] A = $\emptyset$ – множество зависимостей пусто.

[0069] Конечная конфигурация после обработки следующая:

[0070] s = [ROOT] – в стеке один служебный символ;

[0071] b – пуст;

[0072] A – содержит искомое дерево разбора.

[0073] Будем считать s_i, где i = (1, 2, ...), i-м верхним элементом стека, b_i , i = (1, 2, ...) – i-й элемент буфера данных.

[0074] Подход Arc-standard system имеет 4 типа операций:

[0075] SHIFT – удаление b_1 из буфера и добавление его в стек;

[0076] LEFT_ARC – добавляет в A связь от s_1 к s_2, с определенной меткой типа связи, и удаляет s_2 из стека;

[0077] RIGHT_ARC – аналогично LEFT_ARC, только с заменой s_1 и s_2.

[0078] SWAP: возвращает из стека второй элемент в буфер.

[0079] Таким образом, получается всего |T | = $2N_i$+1 возможных действий, где N_i – количество типов меток зависимости. Цель синтаксического парсера **210** текста – выбрать наиболее подходящее действие по данной конфигурации.

[0080] Для обучения искусственной нейронной сети требуется на основе имеющихся данных сгенерировать наиболее подходящую последовательность действий. На каждом шаге конфигурация будет содержать необходимые данные, а действие – ответ.

[0081] Результатом работы синтаксического парсера **210** текста является дерево разбора, где для каждого элемента фразы указан его родитель и тип зависимости. Для оценки результатов работы синтаксического парсера могут использовать следующие метрики:

[0082] Unlabeled Attachment Score (UAS) – отношение количества элементов с правильно указанным родителем, к общему количеству элементов;

[0083] Labeled Attachment Score (LAS) – отношение количества элементов с правильно указанным родителем и типом связи, к общему количеству элементов.

[0084] Пример работы модуля **120** обработки текста может иметь следующий вид:

[0085] Вход: Запрос: "человек хчет взять завтра кридит на 100 млн рубелй"

[0086] Выход:

Атрибут	человек	хочет	взять	завтра	кредит	на	10000000	рублей
word_form форма слова в тексте, после опечаток	человек	None	None	завтра	None	None	10000000	рубль
lemma лемма (начальная форма слова), после опечаток	человек	хотеть	взять	завтра	кредит	на	10000000	рубль
token_t	ANIMACY_T	None	None	TIME_TEMPO	None	None	NUM_T	CCY_TOKE

ype тип токена (число / время)	OKEN			RAL_TOKEN			OKEN	N
token_value форматированное значение токена	None	None	None	Date(01.02.2018)	None	None	10000000	None
is_stop_word слово входит в платформенный список стоп-слов	False	True	False	False	False	True	False	False
dependency_type результат синтаксического парсера	nsubj	root	xcomp	advmod	obj	case	nummod	nmod
head результат синтаксического парсера	2	0	2	3	3	8	8	5

[0087] Пользователь не всегда может точно сформулировать свой запрос на получение информации, которая ему необходима. Более того, даже после получения этой информации требуется ее последующая аналитическая обработка с целью определения ее полезности и пригодности для решения поставленной задачи. Трудности, связанные с решением этой задачи, заключаются в многообразии возможных форм выражения одной и той же идеи, мысли, что особенно характерно для русскоязычных текстов.

[0088] Для решения этой задачи в данном техническом решении формируют векторную модель слов пользовательского ввода. Главным достоинством векторной модели является возможность поиска и ранжирования документов по подобию, то есть по их близости в векторном пространстве за счет определения расстояния между словами.

[0089] Координаты векторов формируются таким образом, чтобы косинусное или евклидово расстояние между векторами близких по смыслу слов было меньше, чем расстояние между векторами далеких по смыслу слов. При этом отдельные компоненты вектора могут отражать некоторую конкретную категорию, например в первой компоненте может содержаться информация о времени (прошлое - настоящее - будущее), во второй о физическом размере (маленький - большой), в третьем о стоимости (дорогой - дешевый) и т.д.

[0090] Затем классифицируют посредством модуля **130** ведения диалога с пользователем по меньшей мере часть пользовательского ввода для формирования ответа.

[0091] Система **100** содержит модуль **130** ведения диалога с пользователем, отвечающий за ведение многоступенчатого диалога с пользователем, повторное использование контекста диалога или использования данных из доступных внешних сервисов.

[0092] Данный модуль **130** используется для того, чтобы отдельные сценарии диалога с пользователем по новым тематикам можно было создавать без единой дополнительной строчки программного кода / с небольшим и несложным объёмом кода, чтобы это решение:

[0093] позволяло не программистам заводить новые тематики;

[0094] было легче в поддержке.

[0095] При этом архитектура модуля **130** позволяет встраивание со своими (совместимыми с архитектурой решения) подпрограммами, которые отвечают отдельным тематикам и могут быть произвольной сложности.

[0096] Для возможности классификации запросов пользователей, а соответственно поддержания контекста беседы с пользователем, опроса клиента по заданному набору

вопросов, модуль **130** может использовать подход на основе алгоритма заполнения слотов или форм (от англ. Slot Filling), широко известный из уровня техники.

[0097] При использовании данного подхода могут определять два типа сценариев:

- линейные сценарии диалога, в которых собирается информация, необходимая для ответа на запрос пользователя из базы данных **150** и/или внешних источников информации, и
- нелинейные сценарии, в которых собираемая информация зависит от предыдущих ответов пользователя.

[0098] При таком подходе каждый линейный сценарий представляется в виде формы, состоящей из набора полей, временем хранения формы и подготовленного ответа. Ответ может представлять собой цепочку произвольных действий и выполняется после заполнения всех обязательных полей формы.

[0099] Каждое поле формы задаётся в виде набора свойств:

- значение по умолчанию;
- уточняющий вопрос для заполнения данного поля;
- обязательность заполнения (если поле не заполнено, но обязательно - будет задан уточняющий вопрос в диалоге пользователю);
- функция для заполнения (одна из заранее заготовленных, которая из результата предобработки текста или использования вспомогательных внешних сервисов/баз данных извлекает значение для заполнения поля);
- возможность использовать значение поля формы из контекста другим формам и возможность заполнить поле значением из контекста;
- время жизни значения поля (после которого оно будет очищено) и другие.

[00100] Нелинейные сценарии при таком подходе задаются в виде однонаправленного ациклического графа форм с указанием начальной для сценария формы. В таком случае все формы являются аналогичными формам линейных сценариев и заполняются аналогично, но между ними существуют возможности переходов. Возможность перехода от заполнения одной формы к другой задаётся в виде накладываемых ограничений на значения полей первой.

[00101] Таким образом, модуль **130** для классификации запросов работает по следующему алгоритму:

[00102] 1. Получает обработанный (шаг **402**) и распознанный (шаг **403**) на предыдущем шаге текст, полученный из модуля **120** предобработки текста;

[00103] 2. Составляет упорядоченный список сценариев для прохода по ним: последние для данного пользователя, а затем все оставшиеся. Каждый сценарий может быть представлен одной формой и относится к одной теме/запросу/необходимой услуге;

[00104] 3. Условия попадания в тот или иной сценарий задаются в виде правил, встраиваемых произвольных классификаторов, примеров, которые используются для классификации платформенным методом, использующимся и в модуле **140** ответа на часто задаваемые вопросы, или их комбинации. Последние для пользователя сценарии проверяются на предмет возможности заполнения полей формы клиентским запросом, и в случае успешности управление передаётся на них. В противном случае может использоваться иерархическая классификация для составления списка наиболее вероятных сценариев проверки, после чего проверяются специфичные для них условия попадания в сценарий;

[00105] 4. Управление передаётся на наиболее вероятный сценарий среди набора сценариев, прошедших пороги уверенности, которые являются необходимыми и специфичными для каждого сценария. Пользователю в таком случае возвращается ответ из этого сценария посредством заполнения формы сценария. В случае отсутствия значимой разницы в вероятности двух прошедших различных сценариев пользователю может быть задан уточняющий вопрос, какой из них подразумевается..

[00106] В тех случаях, когда модуль **130** ведения диалога не включает ни одного сценария, соответствующего запросу пользователя, используется модуль **140** ответа на часто задаваемые вопросы. Данный модуль **140** используется, когда не требуется ни ведения диалога, ни клиентно-зависимых ответов, а только ответы на часто задаваемые вопросы. Для добавления новых тем нужны минимальные усилия: только список самих вопросов, ответов на них и возможных перифразировок вопросов (хотя поиск и допускает произвольность, но большее количество примеров всё же улучшает качество).

[00107] Пример работы:

Вход: "Привет"

Выход: "Привет-привет. :)"

Вход: "что у меня с просрочкой"

Выход: Для уточнения информации о наличии просроченной задолженности Вам необходимо позвонить по № телефона 8-800-333-31-38"

[00108] Необходимые данные для работы данного модуля **140** могут быть следующими:

[00109] эталонное название вопроса для навигации,

[00110] ответ для пользователя,

[00111] каналы передачи данных, в которых этот вопрос доступен,

[00112] позитивные и негативные примеры запросов (позитивные – это запросы, которые должны вести к этому ответу, негативные - похожие на позитивные запросы, которые не должны вести к этому ответу).

[00113] Классификация запроса по пользовательским тематикам из модуля **130** и модуля **140** в некоторых реализациях может быть объединена. После определения тематики управление передаётся в соответствующий модуль.

[00114] В некоторых вариантах реализации для поиска нужного сценария нормализованный текст пользовательского ввода преобразовывают в вектор слов (формируют векторную модель слов) с помощью статистической меры TF-IDF, используемой для оценки важности слова в контексте документа, являющегося частью коллекции документов или корпуса.

[00115] TF-IDF – статистическая мера, используемая для оценки важности слова в контексте документа, являющегося частью коллекции документов или корпуса. Вес некоторого слова пропорционален количеству употребления этого слова в документе, и обратно пропорционален частоте употребления слова в других документах коллекции.

[00116] Мера TF-IDF часто используется в задачах анализа текстов и информационного поиска, например, как один из критериев релевантности документа поисковому запросу, при расчёте меры близости документов при кластеризации.

[00117] TF (term frequency – частота слова) – отношение числа вхождения некоторого слова к общему количеству слов документа. Значимость слова в пределах отдельного документа может быть определена следующей характеристикой:

$$tf(t,d) = \frac{n_i}{\sum_k n_k},$$

[00118] где n_i - число вхождений слова t_i в документ d; $\sum_k n_k$ –общее число слов в данном пользовательском запросе и/или документе.

[00119] IDF (inverse document frequency – обратная частота документа) – величина, обратно пропорциональная частоте, с которой некоторое слово встречается в документах коллекции.

[00120] Учёт IDF уменьшает вес широкоупотребительных слов. Для каждого уникального слова в пределах конкретной коллекции документов существует только одно значение IDF. IDF-характеристика определяется следующим отношением:

[001] $$idf\ (t,D) = log\frac{|D|}{|d_i \supset t_i|}$$

[00121] где $|D|$ – количество документов в корпусе; $|d_i \supset t_i|$ – количество документов, в которых встречается t_i.

[00122] Таким образом, мера TF-IDF является произведением двух сомножителей:

$$tf \cdot idf(t, d, D) = tf(t, d) \times idf(t, D).$$

[00123] Большой вес в мере TF-IDF получают слова с высокой частотой в пределах конкретного документа и с низкой частотой употреблений в других документах.

[00124] Мера TF-IDF часто используется для представления документов коллекции в виде числовых векторов, отражающих важность использования каждого слова из некоторого набора слов (количество слов набора определяет размерность вектора) в каждом документе. Подобная модель называется векторной моделью и даёт возможность сравнивать тексты, сравнивая представляющие их вектора в какой либо метрике (евклидово расстояние, косинусная мера, манхэттенское расстояние, расстояние Чебышева и др.), то есть, производя кластерный анализ.

[00125] В данном техническом решении для формирования выходных данных (шаг **404**, как показано на Фиг. 4) в диалоге может использоваться Word2vec — программный инструмент анализа семантики естественных языков, представляющий собой технологию, которая основана на дистрибутивной семантике и векторном представлении слов.

[00126] Word2vec предполагает нахождение связей между семантикой слов согласно предположению, что слова, находящиеся в похожих контекстах, имеют
тенденцию значить похожие вещи, т.е. быть семантически близкими, как это показано на Фиг. 5. Например, word2vec позволяет использовать «математику» на словах: «король»-«мужчина»+ «женщина» = «королева». Более формально задача стоит так: максимизация косинусной близости между векторами слов (скалярное произведение векторов), которые появляются рядом друг с другом, и минимизация косинусной близости между векторами слов, которые не появляются друг рядом с другом. Рядом друг с другом в данном случае значит в близких контекстах.

[00127] Например, слова «анализ» и «исследование» часто встречаются в похожих контекстах. Фразы «Ученые провели анализ алгоритмов» или «Ученые провели исследование алгоритмов» достаточно похожи. Word2vec анализирует эти контексты и делает вывод, что слова «анализ» и «исследование» являются близкими по смыслу.

[00128] Например, для слова "кофе" word2vec может выдать следующие 15 ближайших соседей в формате "слово - косинусное расстояние":

[00129] коффе 0.734483;

[00130] чая 0.690234;

[00131] чай 0.688656;

[00132] капучино 0.666638;

[00133] кофн 0.636362;

[00134] какао 0.619801;

[00135] эспрессо 0.599390;

[00136] кофя 0.595211;

[00137] цикорий 0.594247;

[00138] кофэ 0.593993;

[00139] копучино 0.587324;

[00140] шоколад 0.585655;

[00141] капучинно 0.580286;

[00142] кардамоном 0.566781;

[00143] латте 0.563224.

[00144] Для формирования ответа пользователю (шаг **404**) в некоторых вариантах реализации изобретения используют векторы всех слов, которые входят в пользовательский ввод данных, определяют между ними среднее (центроид) расстояние, после чего определяют уже косинусное расстояние между словами пользовательского ввода и центроида.

[00145] В некоторых вариантах реализации используют вместо word2vec, такие векторные модели как GloVe, FastText и др., не ограничиваясь.

[00146] В данном техническом решении для векторного представления слов могут использоваться следующие модели CBOW (Continuous Bag-of-Words) и Skip-gram, не ограничиваясь.

[00147] Затем по мультииндексу векторов вопросов из FAQ (база данных часто задаваемых вопросов пользователями), который находится в базе данных **150,** ранжируется несколько максимально похожих по евклидову расстоянию или косинусному сходству. Мультииндекс — обобщение понятия целочисленного индекса до векторного индекса, которое нашло применение в различных областях математики, связанных с функциями многих переменных. Косинусное сходство — это мера сходства между двумя векторами предгильбертового пространства, которая используется для измерения косинуса угла между ними.

[00148] Если даны два вектора признаков (в данном случае два вектора слов), A и B, то косинусное сходство, $\cos(\theta)$, может быть представлено с использованием их скалярного произведения и нормы:

$$\cos(\Theta) = \frac{\mathrm{AgB}}{\|\mathrm{A}\|\|\mathrm{B}\|} = \frac{\sum_{i-1}^{n} \mathrm{A}_i \times \mathrm{B}_i}{\sqrt{\sum_{i-1}^{n} (\mathrm{A}_i)^2} \times \sqrt{\sum_{i-1}^{n} (\mathrm{B}_i)^2}}$$

[00149] В некоторых вариантах реализации для определения максимально похожих вопросов, которые заранее известны, и ответы на них, может использоваться в качестве метрики расстояние Минковского (L1), Евклидово расстояние (L2), Мера Жаккарда.

[00150] Например, если у нас имеется фраза «Джессика любит апельсины и мандарины», которая может быть представлена как вектор слов A (A: [1,1,0,0,0,1,1,1]), и фраза «Джесси нравятся цитрусовые», которая может быть представлена как вектор B (B: [0,0,1,1,1,0,0,0]). В связи с тем, что общих слов нет, косинусное расстояние будет равно 0. Однако нужно учитывать, что смысл в данных фразах семантически похож, так как «цитрусовые» в дереве разбора является родителем слов «апельсины» и «мандарины».

[00151] Далее форматируется результат. В зависимости от расстояния вопроса до лучшего варианта и общего порога отсечения, пользователю выдаётся либо заранее подготовленный ответ (шаг **405**, как показано на Фиг. 4), либо пара (вопрос, ответ) (если шанс нерелевантности ответа достаточен), либо переключаем на оператора.

[00152] Потенциально, система **100** может решить выводить в нужном месте список этих вопросов на выбор пользователя, если это позволяет графический интерфейс пользователя или требует какая-то отдельная команда. Для поддержки такого режима достаточно передать в этот модуль **140** дополнительный параметр, означающий возврат списка лучших вариантов для форматирования вместо самого текста одного.

[00153] В других вариантах реализации изобретения, предварительно формируют перечень часто задаваемых вопросов в базе данных, а затем при поступлении нового пользовательского ввода, применяют способ Word Mover's Distance, предложенный в источнике информации [1].

[00154] При использовании данного подхода находят для каждого слова из одной фразы ближайшего соседа из другой. Word Mover's Distance (WMD) – это минимальное расстояние для перехода из одной фразы в другую, как показано на Фиг. 3.

[00155] Минимальное кумулятивное расстояние по данному подходу может определяться следующим образом.

[00156] Пусть имеется набор векторов слов и расстояний между словами для двух фраз: например, вопрос звучит после предобработки текста как «здравствовать как купить авто кредит», а ответ должен быть «привет давать кредит машина».

	привет	давать	кредит	машина
здравствовать	0.64	0.90	1.11	0.87
как	0.82	1.02	1.19	0.93
купить	0.88	0.91	1.29	0.89
авто	0.80	0.95	1.00	0.39
кредит	1.08	1.26	0.00	1.07

[00157] Далее модуль **140** берет минимальное значение расстояния между словами из каждого столбца, из каждой строки, после чего их суммируют, делят на общее количество слов (нормализуют) и получают «стоимость перехода из одной фразы в другую»:

[00158] (0.64 + 0.90 + 0.00 + 0.39 + 0.64 + 0.82 + 0.88 + 0.39 + 0.00) / 9 = 0.52

[00159] В некоторых вариантах реализации могут осуществлять нормализацию на общее количество уникальных токенов или на количество уникальных токенов во фразе, не ограничиваясь.

[00160] В некоторых вариантах реализации для темы «Как заблокировать карты» может ошибочно подходить «Я залочил кредитку, как отменить?». В таких случаях хранят в базе данных **150** такие негативные примеры, их связь между собой по семантике, и учитывают это в дальнейшей работе.

[00161] В некоторых вариантах реализации изобретения используют не только расстояние между словами, но и дополнительные атрибуты, включая, но не ограничиваясь: частоту встречаемости каждого слова в базе данных **150**, морфологические признаки каждого слова, синтаксическую информацию.

[00162] В некоторых вариантах реализации изобретения используют не только расстояние между словами, но и некоторый собственный вес каждого слова, например, частоту его встречаемости.

[00163] Если x - доля всех тем, в которых встречается слово W, а y - доля всех примеров в данной теме, в которых встречается слово W, домножают расстояния между словами на $1 - x^y$ для осуществления нормализации.

[00164] Также проблема традиционного подхода WMD может заключаться в том, что он не учитывает порядок слов, так что фразы «банкомат съел карту» и «карта съела

банкомат» идентичны после лемматизации. Для решения данной проблемы могут использовать синтаксическую информацию из синтаксического парсера **210** текста, который отображает дерево разбора слов. В конкретном варианте реализации синтаксический парсер **210** отображает ошибку, так как конструкция «карта съела банкомат» не может быть использована. Для исправления данной ситуации вводят матрицу стоимости переходов из одного класса в другой и домножают евклидовы расстояния между фразами на соответствующие значения.

[00165] Для обеспечения возможности делать антонимы дальше по расстоянию между словами, а синонимы ближе в word2vec с точки зрения например евклидова расстояния, не сильно повреждая при этом расстояния остальных слов в пространстве, используют подходы описанные в уровне техники (источник информации [2]), например, Retrofitting.

[00166] Если ни на одном из предыдущих шагов (функционирование модуля **130** и модуля **140**) не удалось подобрать ответ пользователю в диалоге графического интерфейса пользователя, то пользователю будет выдано сообщение об этом, либо пользователь будет переключен на оператора. В некоторых случаях может произойти предложение переключить пользователя на оператора для связи по другому каналу передачи данных.

[00167] В конкретном варианте реализации, не ограничиваясь, может быть выбран режим работы, симулирующий отсутствие самого чат бота и плавно переводящий на оператора при практически любом непонятном запросе пользователя (так сделано из-за незначительного объёма контента чат-бота):

[00168] если чат-бот ожидает ответа на заданный им вопрос и получает сообщение, которое не может распознать модуль **130** или модуль **140**, то он даёт ещё одну попытку пользователю (сообщая ему об этом) и в случае повторения переводит на оператора;

[00169] если чат бот не ожидает ответа, а находится просто в ожидании запроса от пользователя и получает нераспознаваемое сообщение, то сразу переводит пользователя на оператора.

[00170] В целом же, поведение модуля настраивает специальный параметр вида v = [2,3,6]. На заданном примере чат бот себя будет вести следующим образом:

[00171] при получении 1-го нераспознанного сообщения факт его существования забывается после v[1] = 2 распознанных сообщений;

[00172] при получении 2-го нераспознанного сообщения факт его существования забывается после v[2] = 3 распознанных сообщений;

[00173] при получении 3-го нераспознанного сообщения факт его существования забывается после v[3] = 6 распознанных сообщений;

[00174] получение 4-го нераспознанного сообщения переводит пользователя на оператора (или, более мягко, предлагает ему это сделать в графическом интерфейсе пользователя).

[00175] В общем случае, если перевод на оператора не требуется, выводится одно случайное сообщение из заготовок с единым смыслом: сообщается пользователю, что ничего не найдено.

[00176] Иллюстративные варианты реализации, описанные здесь, могут воплощаться в операционной среде, содержащей выполняемые компьютером команды (например, программное обеспечение), установленной на компьютер, в аппаратных средствах, или в комбинации программного обеспечения и аппаратных средств. Выполняемые компьютером команды могут быть написаны на компьютерном языке программирования или могут воплощаться в аппаратной логике. Если написаны на языке программирования в соответствии с признанным стандартом, такие команды могут выполняться на множестве аппаратных платформ и для интерфейсов множества операционных систем. Хотя не ограничиваясь этим, компьютерные программы системы программного обеспечения для воплощения настоящего способа могут быть написаны на любом количестве подходящих языков программирования, таких как, например, язык гипертекстовой разметки (HTML), динамический HTML, расширяемый язык разметки (XML), расширяемый язык таблиц стилей (XSL), язык семантики стиля и спецификаций документов (DSSSL), каскадные таблицы стилей (CSS), язык интеграции синхронизированных мультимедийных данных (SMIL), язык разметки документов для беспроводной связи (WML), JavaTM, JiniTM, C, C++, Perl, UNIX Shell, Visual Basic или Visual Basic Script, язык разметки событий виртуальной реальности (VRML), ColdFusionTM или другие компиляторы, ассемблеры, интерпретаторы или другие компьютерные языки или платформы.

[00177] Система **100** может быть реализована посредством архитектуры, как показано на Фиг. 6, и включать следующие компоненты, показанные ниже, в том числе процессор **610**. В конкретном варианте реализации настоящего технического решения, процессор **610** может включать в себя один или несколько процессоров и/или один или несколько микроконтроллеров, выполненных с возможностью выполнять инструкции для выполнения операций, связанных с работой вышеупомянутого способа для выстраивания диалога с пользователем в удобном для пользователя канале. В различных вариантах реализации настоящего технического решения, процессор **610** может быть реализован в виде однокристальных, многокристальных и/или электрических компонентов, включая одну или несколько интегральных схем и печатных плат. Процессор **610** может

опционально содержать блок кэш-памяти (не показан) для временного локального хранения инструкций, данных или компьютерных адресов. Например, процессор **610** может включать в себя один или несколько процессоров или один или несколько контроллеров, относящихся к конкретным задачам или единый многофункциональный процессор или контроллер.

[00178] Процессор **610** оперативно связан с модулем **620** ввода-вывода данных, аудиомодулем **630**.

[00179] В представленном варианте реализации настоящего технического решения, модуль **620** ввода-вывода данных может быть реализован в виде сенсорного экрана, который выполняет функциональность как устройства ввода (путем фиксирования пользовательских команд в виде прикосновений), так и устройства пользовательского вывода (т.е. дисплея). Другими словами, сенсорный экран представляет собой дисплей, который определяет наличие и положение пользовательского ввода-прикосновения. В альтернативных вариантах реализации настоящего технического решения, модуль **620** ввода-вывода может быть реализован как отдельный дисплей и отдельное устройство ввода. Тем не менее, в других альтернативных вариантах реализации настоящего технического решения, модуль **620** ввода-вывода может включать в себя физическую клавиатуру (содержащую одну или несколько физических кнопок) в дополнение к сенсорному экрану.

[00180] Процессор **610** дополнительно связан с модулем **640** памяти, которая содержит базу данных **150**. Модуль **640** памяти может охватывать один или несколько носителей и в целом предоставлять место для хранения компьютерного кода для реализации вышеупомянутого способа передачи информации о p2p-переводе (например, программного и/или аппаратного обеспечения). Например, модуль **640** памяти может включать в себя различные материальные машиночитаемые носители, включая постоянное запоминающее устройство (ПЗУ) и/или оперативное запоминающее устройство (ОЗУ). Как известно специалистам в данной области техники, ПЗУ однонаправленно передает данные и инструкции процессору **610**, а ОЗУ обычно используется для передачи данных и инструкций в двустороннем порядке.

[00181] Модуль **640** памяти также может включать в себя одно или несколько фиксированных устройств хранения данных в форме, например, жесткого диска (HDD), твердотельного накопителя (SSD), карты флеш-памяти (например, Secured Digital или SD-карты, мультимедийной карты eMMD), наряду с другими видами памяти, двусторонне соединенными с процессором **610**. Информация может также располагаться на одном или нескольких съемных носителях, загруженных или установленных в системе **100**, когда это

необходимо. Например, любая из ряда подходящих карт памяти (например, SD-карт) может быть загружена в систему **100** на временной или постоянной основе (с помощью, например, одного или нескольких набором дополнительных портов).

[00182] Модуль **640** памяти может хранить среди прочего серию машиночитаемых инструкций, при выполнении которых процессор **610** (а также другие компоненты системы **100**) настраиваются на выполнение различных операций, описанных здесь.

[00183] В различных конкретных вариантах реализации, система **100** может дополнительно содержать модуль **650** беспроводной связи и сенсорный модуль **660**, оба из которых соединены с процессором **610** для упрощения различных функций системы **100**.

[00184] Модуль **650** беспроводной связи может быть создан для работы через одну или несколько беспроводных сетей, например, беспроводную персональную сеть (WPAN) (такую как, BLUETOOTH WPAN, ИК персональная сеть), WI-FI сеть (например, 802.11a/b/g/n WI-FI сеть, сеть стандартов 802.11), WI-MAX сеть, мобильную сотовую сеть. В качестве мобильной сотовой сети может использоваться, например, сеть глобальной системы мобильной связи (GSM), сеть развитого стандарта GSM с увеличенной скоростью передачи данных (EDGE), сеть универсальной мобильной телекоммуникационной системы (UMTS) или сеть долговременного развития (LTE). Дополнительно, модуль **650** беспроводной связи может включать в себя хостинг протоколы, таким образом, что система **100** может быть выполнена как базовая станция для беспроводных устройств.

[00185] Сенсорный модуль **660** может включать в себя одно или несколько сенсорных устройств, чтобы предоставлять дополнительный ввод и упрощать различные функции системы **100**. Некоторые примеры вариантов реализации сенсорного модуля **660** могут включать в себя одно или несколько из устройств: акселерометр, устройство для измерения температуры окружающей среды, устройство для измерения силы гравитации, гироскоп, устройство для измерения освещенности, устройство для измерения силы ускорения, устройство для измерения окружающего геомагнитного поля, устройство для измерения степени вращения, устройство для измерения атмосферного давления, устройство для измерения относительной влажности, устройство для измерения ориентации устройства и так далее. Следует отметить, что некоторые из этих устройств могут быть реализованы как аппаратное обеспечение, программное обеспечение или комбинация их обоих.

[00186] Также предлагается модуль **670** источника питания для предоставления питания одному или нескольким компонентам системы **100**. В некоторых вариантах осуществления настоящего технического решения, модуль **670** источника питания может

быть реализован как литий-ионный аккумулятор. Тем не менее, могут быть использованы другие типы аккумуляторных (и обычных) батареек. Естественно, в других вариантах осуществления настоящего технического решения дополнительно или альтернативно к использованию батареи, модуль **670** источника питания может быть реализован как главный источник питания, выполненный с возможностью присоединения системы **100** к главному источнику питания, например, стандартному кабелю питания и вилке.

[00187] В некоторых вариантах осуществления настоящего технического решения, различные компоненты системы **100** могут быть соединены друг с другом через одну или несколько шин (включая аппаратное и/или программное обеспечение), эти шины не пронумерованы. В качестве неограничивающего примера, одна или несколько шин могут включать в себя ускоренный графический порт (AGP) или другие графические порты, улучшенную архитектуру шины промышленного стандарта (EISA), переднюю шину (FSB), гипертранспортную шину (HT), шину промышленной стандартной архитектуры (ISA), соединение INFINIBAND, LPC-шину, шину памяти, шину микроканальной архитектуры (MCA), шину соединения периферийных компонентов (PCI), шину соединения периферийных компонентов типа экспресс (PCI-X), шину последовательного интерфейса обмена данными с накопителями информации (SATA), локальную шину ассоциации видеоэлектронных стандартов (VLB), интерфейс универсального асинхронного приемопередатчика (UART), последовательную шину данных для связи интегральных схем (I2C), шину последовательного периферийного интерфейса (SPI), интерфейс памяти Secure Digital (SD), а интерфейс памяти MultiMediaCard (MMC), интерфейс памяти Memory Stick (MS), интерфейс Secure Digital Input Output (SDIO), шину многоканального буферизованного последовательного порта (McBSP), универсальную последовательную шину (USB), шину контроллера универсального запоминающего устройства (GPMC), шину контроллера синхронной динамической памяти с произвольным доступом (SDRC), шину ввода/вывода общего назначения (GPIO), шину раздельного видеосигнала (S-Video), шину последовательного интерфейса дисплея (DSI), шину расширенной шинной архитектуры для микроконтроллеров (AMBA), или любую другую подходящую шину или комбинацию двух или более шин.

[00188] Система **100** в некоторых вариантах реализации при помощи модулей «speech-to-text» и «text-to-speech» может быть использована для голосового обслуживания (IVR) и/или голосового управления мобильным приложением.

[00189] Специалисты в данной области техники поймут, что в настоящем описании выражение «получение данных» от пользователя подразумевает получение электронным устройством данных от пользователя в виде электронного (или другого) сигнала. Кроме

того, специалисты в данной области техники поймут, что отображение данных пользователю через компонент графического интерфейса пользователя (например, экран электронного устройства и тому подобное) может включать в себя передачу сигнала компоненту графического интерфейса пользователя, этот сигнал содержит данные, которые могут быть обработаны, и по меньшей мере часть этих данных может отображаться пользователю через компонент графического интерфейса пользователя.

[00190] Некоторые из этих этапов, а также передача-получение сигнала хорошо известны в данной области техники и поэтому для упрощения были опущены в конкретных частях данного описания. Сигналы могут быть переданы-получены с помощью оптических средств (например, оптоволоконного соединения), электронных средств (например, проводного или беспроводного соединения) и механических средств (например, на основе давления, температуры или другого подходящего параметра).

[00191] Модификации и улучшения вышеописанных вариантов осуществления настоящего технического решения будут ясны специалистам в данной области техники. Предшествующее описание представлено только в качестве примера и не несет никаких ограничений. Таким образом, объем настоящего технического решения ограничен только объемом прилагаемой формулы изобретения.

ИСПОЛЬЗУЕМЫЕ ИСТОЧНИКИ ИНФОРМАЦИИ

[00192] 1. Kusner M. et al. From word embeddings to document distances //International Conference on Machine Learning. – 2015. – С. 957-966.

[00193] 2. Faruqui M. et al. Retrofitting word vectors to semantic lexicons //arXiv preprint arXiv:1411.4166. – 2014.

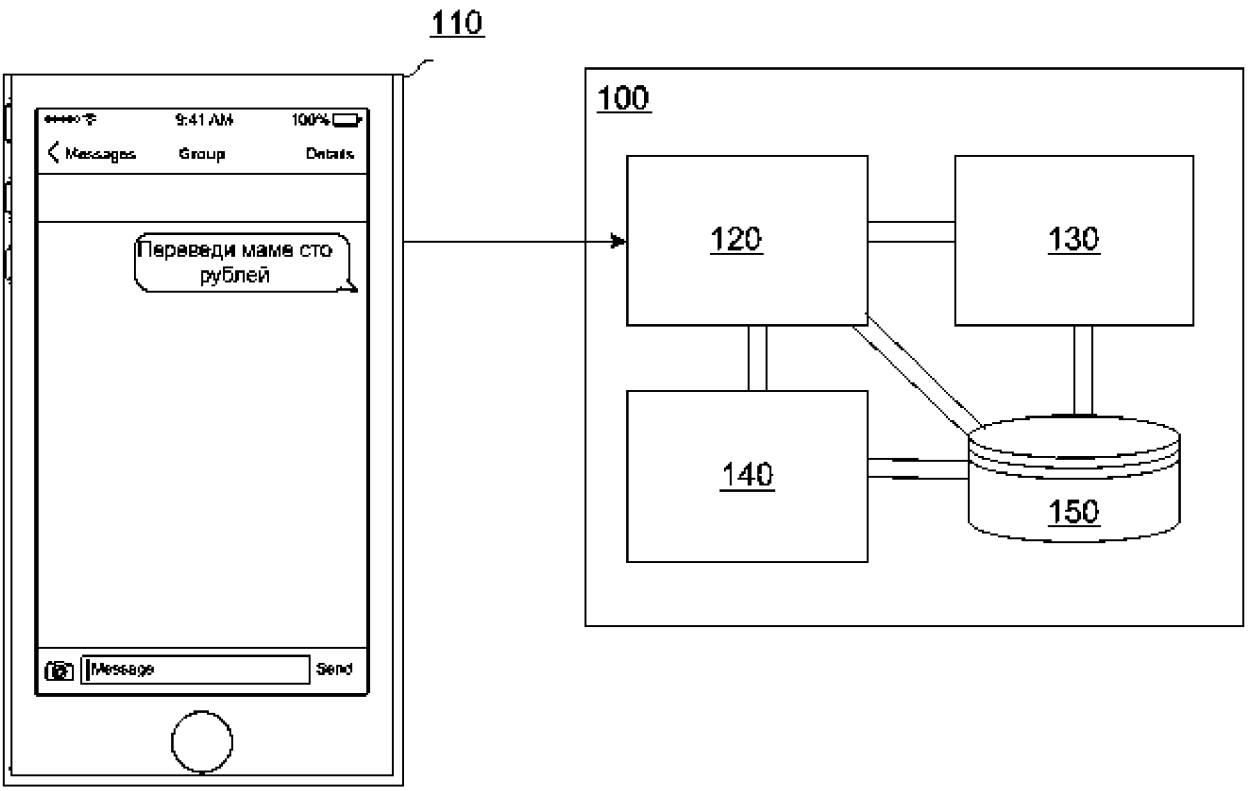
[00194] 3. Пруцков А. В. Генерация и определения форм слов естественных языков на основе их последовательных преобразований //Вестник Рязанского государственного радиотехнического университета. – 2009. – №. 27. – С. 51.

ФОРМУЛА

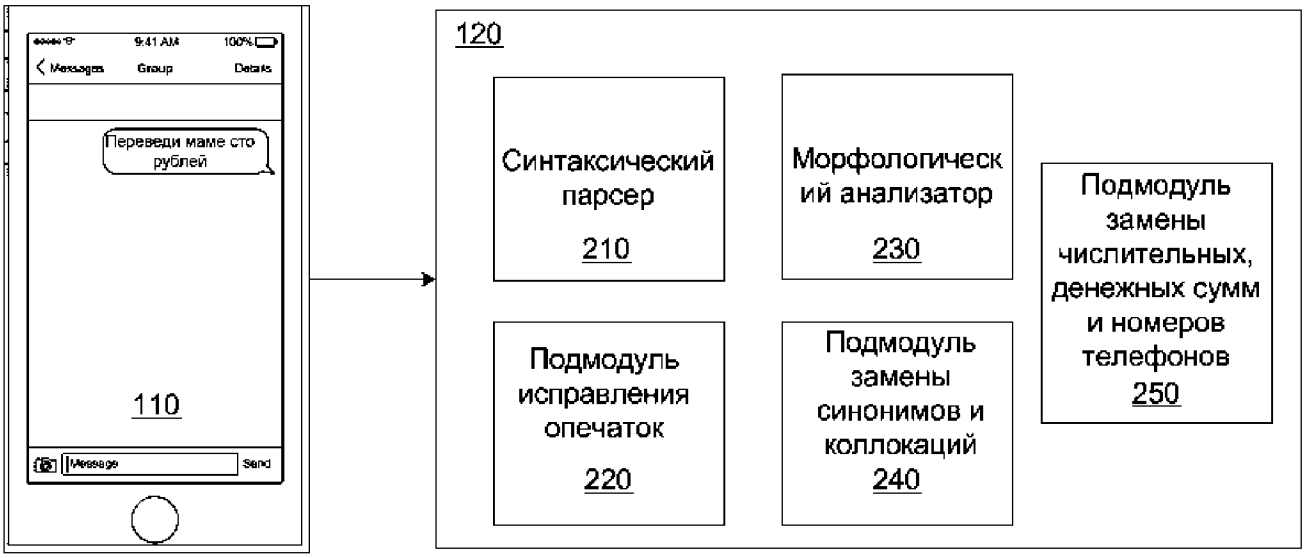
1. Способ для выстраивания диалога с пользователем в удобном для пользователя канале, включающий следующие шаги:
    - получают посредством процессора, функционально соединенного с базой данных, данные пользовательского ввода;
    - осуществляют предобработку пользовательского ввода посредством его разбиения на предложения и слова, причем
        - исправляют орфографические ошибки пользователя посредством использования модуля исправления опечаток;
        - выполняют лемматизацию каждого слова из пользовательского ввода;
        - формируют структуру зависимостей слов друг от друга в пользовательском вводе посредством использования синтаксического парсера;
    - формируют векторную модель слов пользовательского ввода;
    - классифицируют посредством модуля ведения диалога с пользователем по меньшей мере часть пользовательского ввода для формирования ответа;
    - формируют ответ для пользователя для отображения в графическом интерфейсе пользователя;
    - предоставляют посредством процессора по меньшей мере один ответ на распознанный пользовательский ввод данных.
2. Способ по п. 1, характеризующийся тем, что при осуществлении предобработки пользовательского ввода дополнительно исправляют опечатки в пользовательском вводе посредством подмодуля исправления опечаток.
3. Способ по п. 1, характеризующийся тем, что при осуществлении предобработки пользовательского ввода дополнительно выполняют токенизацию числительных из пользовательского ввода.
4. Способ по п. 1, характеризующийся тем, что синтаксический парсер текста для анализа пользовательского текста использует рекуррентные нейронные сети.

5. Способ по п. 1, характеризующийся тем, что при классификации пользовательского ввода модуль ведения диалога использует алгоритм заполнения слотов.
6. Способ по п. 1, характеризующийся тем, что при классификации посредством модуля ведения диалога с пользователем пользовательского ввода определяют линейные сценарии диалога или нелинейные сценарии диалога.
7. Способ по п. 1, характеризующийся тем, что при формировании векторной модели слов пользовательского ввода используют модель CBOW и/или Skip-gram.
8. Способ по п. 1, характеризующийся тем, что при формировании ответа для пользователя для отображения в графическом интерфейсе пользователя определяют класс, к которому относится пользовательский ввод.
9. Способ по п. 1, характеризующийся тем, что при формировании ответа для пользователя для отображения в графическом интерфейсе пользователя в случае отсутствии ответа в диалоге, переключают его на оператора.
10. Система для выстраивания диалога с пользователем в удобном для пользователя канале, содержащая:
    - модуль предобработки текста, выполненный с возможностью получения пользовательского ввода и его лемматизации, исправления в нем опечаток, токенизации и синтаксическом анализе;
    - модуль ведения диалога с пользователем, выполненный с возможностью получения из модуль предобработки текста обработанного текста пользовательского ввода и классификации, по меньшей мере, части пользовательского ввода для формирования ответа;
    - модуль ответа на часто задаваемые вопросы, выполненный с возможностью получения пользовательского ввода из модуля ведения диалога с пользователем и формирования ответа для пользователя для отображения в графическом интерфейсе пользователя.

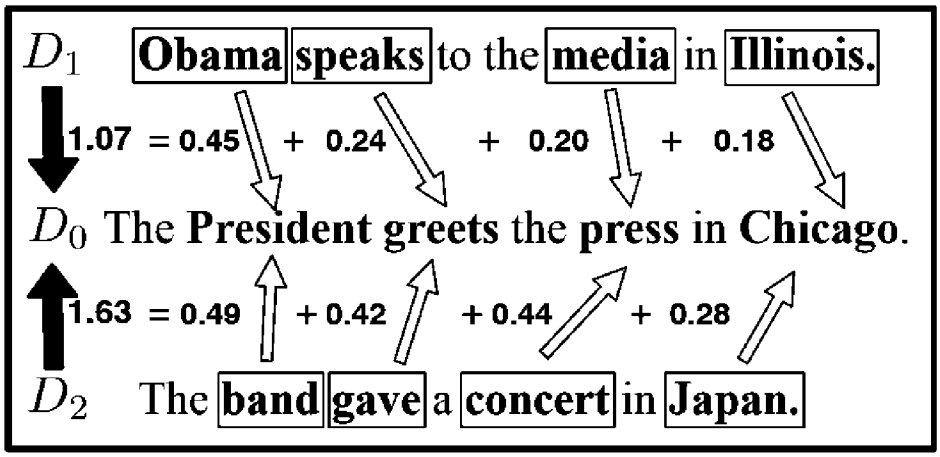
ЧЕРТЕЖИ К ОПИСАНИЮ



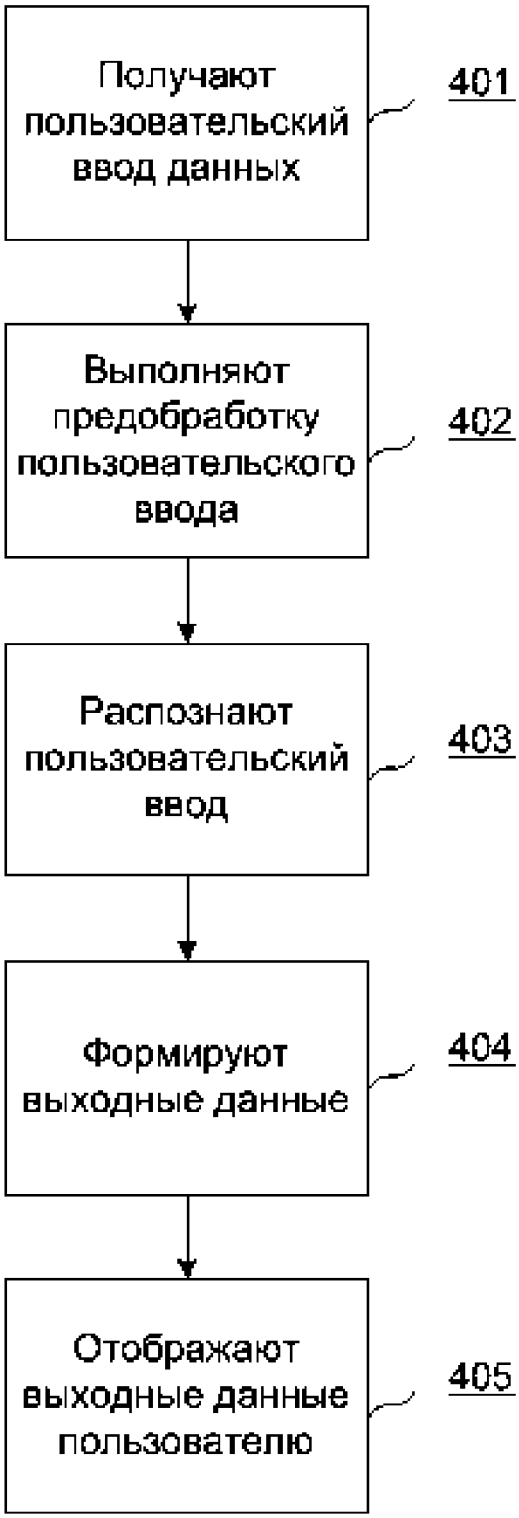
Фиг. 1


9:41 AM
100%
Messages
Group
Details
Переведи маме сто рублей
110
Message
Send
120
Синтаксический парсер
210
Морфолопический анализатор
230
Подмодуль замены числительных, денежных сумм и номеров телефонов
250
Подмодуль исправления опечаток
220
Подмодуль замены синонимов и коллокаций
240

Фиг. 2

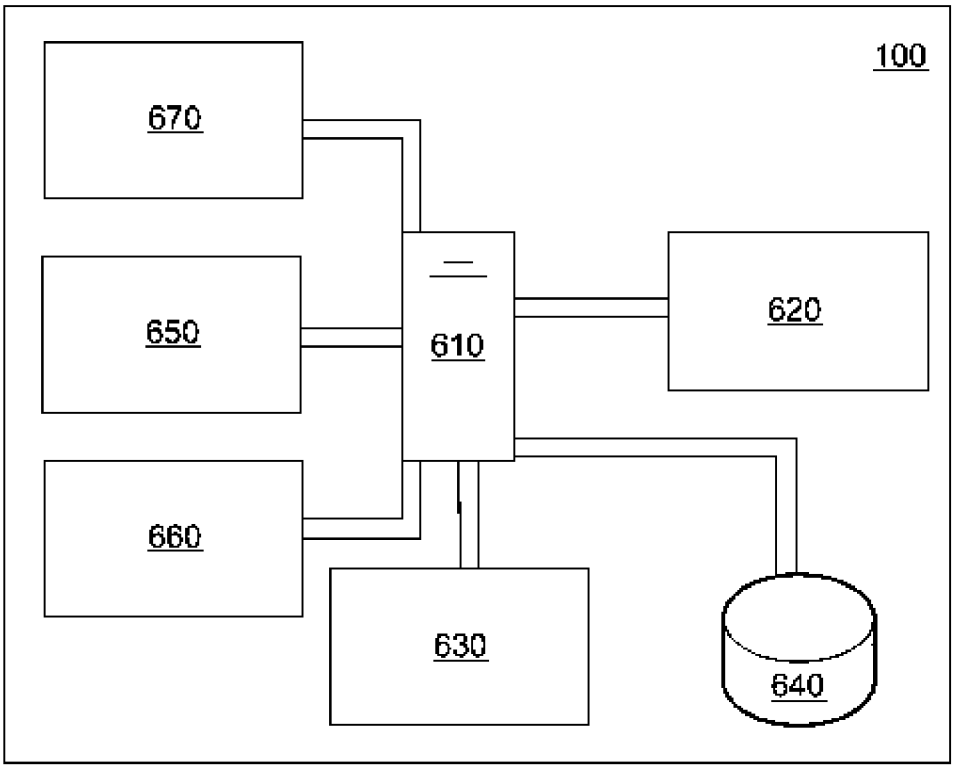

D_1 Obama speaks to the media in Illinois.
1.07 = 0.45 + 0.24 + 0.20 + 0.18
D_0 The President greets the press in Chicago.
1.63 = 0.49 + 0.42 + 0.44 + 0.28
D_2 The band gave a concert in Japan.

Фиг. 3


Получают пользовательский ввод данных
401
Выполняют предобработку пользовательского ввода
402
Распознают пользовательский ввод
403
Формируют выходные данные
404
Отображают выходные данные пользователю
405

Фиг. 4

Фиг. 5



Фиг. 6

ЕВРАЗИИСКОЕ ПАТЕНТНОЕ ВЕДОМСТВО

ОТЧЕТ О ПАТЕНТНОМ ПОИСКЕ

(статья 15(3) ЕАПК и правило 42 Патентной инструкции к ЕАПК)

Номер евразийской заявки: 201891077

Дата подачи: 31 мая 2018 (31.05.2018) | Дата испрашиваемого приоритета: 31 мая 2018 (31.05.2018)

Название изобретения: Способ и система для выстраивания диалога с пользователем в удобном для пользователя канале

Заявитель: ПУБЛИЧНОЕ АКЦИОНЕРНОЕ ОБЩЕСТВО "СБЕРБАНК РОССИИ" (ПАО СБЕРБАНК)

☐ **Некоторые пункты формулы не подлежат поиску (см. раздел I дополнительного листа)**

☐ **Единство изобретения не соблюдено (см. раздел II дополнительного листа)**

А. КЛАССИФИКАЦИЯ ПРЕДМЕТА ИЗОБРЕТЕНИЯ:

МПК: ***G10L 15/22*** *(2013.01)*; ***G06F 17/20*** *(2006.01)*

СПК: ***G10L 15/22*** *(2013-01)*; ***G10L 2015/225*** *(2013-01)*; ***G06F 17/20*** *(2013-01)*; ***G06F 17/27*** *(2013-01)*

Согласно Международной патентной классификации (МПК) или национальной классификации и МПК

Б. ОБЛАСТЬ ПОИСКА:

Минимум просмотренной документации (система классификации и индексы МПК)
G06F 3/00, 3/048-3/08, 3/16, 17/00-17/50, G10L 3/00-13/10, 15/00-15/34

Другая проверенная документация в той мере, в какой она включена в область поиска:

В. ДОКУМЕНТЫ, СЧИТАЮЩИЕСЯ РЕЛЕВАНТНЫМИ

Категория*	Ссылки на документы с указанием, где это возможно, релевантных частей	Относится к пункту №
Y	EP 2650776 A1 (ARTIFICIAL SOLUTIONS IBERIA, S.L.) 16.10.2013, реферат, параграфы [0003], [0005]-[0006], [0008]-[0010], [0039]-[0041], [0044]-[0045], [0048]-[0050], [0057]-[0096], [0104]-[0108], [0113], [0116]-[0142], [0145]-[0147], п.п. 1-3 формулы, фиг. 1-3	1-10
Y	US 2017/0053646 A1 (MITSUBISHI ELECTRIC RESEARCH LABORATORIES, INC.) 23.02.2017, реферат, параграфы [0002]-[0004], [0011], [0014]-[0017], [0021], [0023]-[0025], [0035]-[0042], [0050], [0052]-[0053], [0058]-[0060], п.п. 1, 4, 6 формулы фиг. 2-3	1-10
A	US 2017/0316775 A1 (CONDUENT BUSINESS SERVICES, LLC) 02.11.2017	1-10
A	US 2015/0032443 A1 (YAEL KAROV et al.) 29.01.2015	1-10
A	US 2017/0358304 A1 (APPLE INC.) 14.12.2017	1-10

☐ последующие документы указаны в продолжении графы В ☐ данные о патентах-аналогах указаны в приложении

* Особые категории ссылочных документов:

"A" документ, определяющий общий уровень техники
"E" более ранний документ, но опубликованный на дату подачи евразийской заявки или после нее
"O" документ, относящийся к устному раскрытию, экспонированию и т.д.
"P" документ, опубликованный до даты подачи евразийской заявки, но после даты испрашиваемого приоритета
"D" документ, приведенный в евразийской заявке
"T" более поздний документ, опубликованный после даты приоритета и приведенный для понимания изобретения
"X" документ, имеющий наиболее близкое отношение к предмету поиска, порочащий новизну или изобретательский уровень, взятый в отдельности
"Y" документ, имеющий наиболее близкое отношение к предмету поиска, порочащий изобретательский уровень в сочетании с другими документами той же категории
"&" документ, являющийся патентом-аналогом
"L" документ, приведенный в других целях

Дата действительного завершения патентного поиска: 12 ноября 2018 (12.11.2018)

Наименование и адрес Международного поискового органа:
Федеральный институт промышленной собственности
РФ, 125993,Москва, Г-59, ГСП-3, Бережковская наб., д. 30-1.Факс: (499) 243-3337, телетайп: 114818 ПОДАЧА

Уполномоченное лицо :
А.Р. Комарова
Телефон № (499) 240-25-91