

(19)



**Евразийское
патентное
ведомство**

(11) **042834**

(13) **B1**

(12) **ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ЕВРАЗИЙСКОМУ ПАТЕНТУ**

(45) Дата публикации и выдачи патента
2023.03.29

(21) Номер заявки
202192280

(22) Дата подачи заявки
2021.09.15

(51) Int. Cl. **G10L 15/08** (2006.01)
G10L 17/18 (2013.01)
G06F 40/00 (2020.01)

(54) **СПОСОБ И УСТРОЙСТВО ПРЕДОСТАВЛЕНИЯ КЛИЕНТУ СЕРВИСА
ПОСРЕДСТВОМ ВИРТУАЛЬНОГО АССИСТЕНТА**

(31) **2021119782**

(32) **2021.07.06**

(33) **RU**

(43) **2023.01.31**

(56) **US-B2-10785270**
US-A1-2021136205
US-A1-2021158234
CN-A-107274889

(71)(73) Заявитель и патентовладелец:
**ПУБЛИЧНОЕ АКЦИОНЕРНОЕ
ОБЩЕСТВО "СБЕРБАНК
РОССИИ" (ПАО СБЕРБАНК) (RU)**

(72) Изобретатель:
**Оберемок Надежда Леонидовна,
Иванов Максим Сергеевич, Сивков
Егор Сергеевич, Нефедов Михаил
Анатольевич (RU)**

(74) Представитель:
Герасин Б.В. (RU)

(57) Изобретение относится в общем к области обработки цифровых данных, а в частности к способу и устройству предоставления клиенту по меньшей мере одного сервиса посредством виртуального (цифрового) ассистента. Техническим результатом, достигаемым при решении вышеуказанной технической проблемы или технической задачи, является расширение функциональных возможностей за счет определения того, что клиент является представителем юридического лица и предоставления ему соответствующего сервиса. Указанный технический результат достигается благодаря осуществлению способа предоставления клиенту по меньшей мере одного сервиса, выполняемого по меньшей мере одним вычислительным устройством, содержащего этапы, на которых получают запрос на предоставление по меньшей мере одного сервиса; преобразуют полученный запрос в текстовую информацию; классифицируют текстовую информацию посредством модели, содержащей два классификатора, причем один классификатор обучен на размеченных текстовых данных, содержащих запросы на предоставление сервисов, предоставляемых физическим лицам (ФЛ), а второй классификатор обучен на размеченных текстовых данных, содержащих запросы на предоставление сервисов, предоставляемых юридическим лицам (ЮЛ); на основе полученного класса определяют, что для предоставления сервиса клиенту необходимо быть представителем ЮЛ; определяют, что клиент является представителем ЮЛ; на основе класса определяют сервис, который следует предоставить клиенту и предоставляют помянутый сервис клиенту.

042834
B1

042834
B1

Область техники

Изобретение относится в общем к области обработки цифровых данных, а в частности к способу и устройству предоставления клиенту по меньшей мере одного сервиса посредством виртуального (цифрового) ассистента.

Уровень техники

На данный момент на рынке представлены виртуальные ассистенты, которые воспринимают любого своего пользователя как физическое лицо без возможности работы с ним как с представителем юридического лица. Например, известны системы и способы предоставления виртуального ассистента, описанные в документе US 10534623 B2, опубл. 14.01.2020. В данном документе раскрывается способ, выполняемый по меньшей мере одним компьютером, включающий использование по меньшей мере одного компьютера для выполнения действий по доступу к информации, определяющей по меньшей мере одно заданное пользователем условие, и по меньшей мере одно соответствующее заданное пользователем действие, которое должно быть выполнено при соблюдении указанного пользователем условия; определение, выполняется ли по меньшей мере одно указанное пользователем условие; и когда определено, что по меньшей мере одно указанное пользователем условие выполнено, инициируют виртуального помощника, выполняющего на мобильном устройстве, отличном от по меньшей мере одного компьютера, выполнять по меньшей мере одно указанное пользователем действие.

Проблематика существующих решений заключается в восприятии текущими реализациями голосовых и текстовых ассистентов каждого клиента исключительно как физическое лицо, что ведет к невозможности предоставления услуг и сервисов по консультированию в рамках голосовых и текстовых сценариев представителям юридических лиц.

Новизна представленного решения заключается в реализации алгоритма, позволяющего объединить процессы взаимодействия виртуального ассистента с физическим лицом и представителем юридического лица и его адаптации в зависимости от роли пользователя, для которой этот сценарий должен быть иницирован.

Сущность технического решения

Технической проблемой или технической задачей, поставленной в данном техническом решении, является создание нового эффективного, простого и безопасного решения для предоставления клиенту по меньшей мере одного сервиса посредством виртуального ассистента.

Техническим результатом, достигаемым при решении вышеуказанной технической проблемы или технической задачи, является расширение функциональных возможностей за счет определения того, что клиент является представителем юридического лица и предоставления ему соответствующего сервиса.

Указанный технический результат достигается благодаря осуществлению способа предоставления клиенту по меньшей мере одного сервиса, выполняемого по меньшей мере одним вычислительным устройством, содержащего этапы, на которых:

получают запрос на предоставление по меньшей мере одного сервиса;

преобразуют полученный запрос в текстовую информацию;

классифицируют текстовую информацию посредством модели, содержащей два классификатора, причем один классификатор обучен на размеченных текстовых данных, содержащих запросы на предоставление сервисов, предоставляемых физическим лицам (ФЛ), а второй классификатор обучен на размеченных текстовых данных, содержащих запросы на предоставление сервисов, предоставляемых юридическим лицам (ЮЛ);

на основе полученного класса определяют, что для предоставления сервиса клиенту необходимо быть представителем ЮЛ;

определяют, что клиент является представителем ЮЛ;

на основе класса определяют сервис, который следует предоставить клиенту, и предоставляют помянутый сервис клиенту.

В одном частном примере осуществления способа дополнительно выполняют этапы, на которых:

преобразуют текстовую информацию в последовательность токенов;

преобразуют последовательность токенов в численный вектор фиксированной размерности.

В другом частном примере осуществления способа дополнительно выполняют этап, на котором определяют, что для ЮЛ, представителем которого является клиент, предоставление запрошенного сервиса разрешено. В другом частном примере осуществления способа дополнительно выполняют этапы, на которых:

извлекают набор сущностей и их атрибуты из текстовой информации;

уточняют параметры сервиса, который будет предоставлен клиенту, на основе извлеченного набора сущностей и их атрибутов.

В другом частном примере осуществления способа дополнительно выполняют этапы, на которых:

на основе набора сущностей определяют контекст текстовой информации;

сравнивают набор сущностей с заданным набором сущностей для данного контекста текстовой информации для определения отсутствующих в текстовой информации сущностей;

дополняют извлеченный набор сущностей отсутствующими сущностями на основе контекста тек-

стовой информации.

В другом предпочтительном варианте осуществления заявленного решения представлено устройство предоставления клиенту по меньшей мере одного сервиса, содержащее по меньшей мере одно вычислительное устройство и по меньшей мере одно устройство памяти, содержащее машиночитаемые инструкции, которые при их исполнении по меньшей мере одним вычислительным устройством выполняют вышеуказанный способ.

Краткое описание чертежей

Признаки и преимущества настоящего технического решения станут очевидными из приводимого ниже подробного описания технического решения и прилагаемых чертежей, на которых:

На фиг. 1 представлена схема системы обработки данных.

На фиг. 2 пример общего вида вычислительного устройства.

Осуществление технического решения

Ниже будут описаны понятия и термины, необходимые для понимания данного технического решения.

В данном техническом решении под системой подразумевается, в том числе компьютерная система, ЭВМ (электронно-вычислительная машина), ЦПУ (числовое программное управление), ПЛК (программируемый логический контроллер), компьютеризированные системы управления и любые другие устройства, способные выполнять заданную, четко определенную последовательность операций (действий, инструкций).

Под устройством обработки команд подразумевается электронный блок, вычислительное устройство, либо интегральная схема (микроспроцессор), исполняющая машинные инструкции (программы).

Устройство обработки команд считывает и выполняет машинные инструкции (программы) с одного или более устройств 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.

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

Виртуальный ассистент - компьютерная программа, имитирующая поведение человека при общении с одним или несколькими собеседниками на естественном языке (голосом или в тексте).

Токенизация - разделение текстовой последовательности на семантически значимые куски (токены).

Векторизация - кодирование отдельных токенов или последовательности токенов в виде упорядоченных множеств чисел (векторов).

Классификация - отнесение векторизованного текста к одному из заранее определенных классов с помощью модели машинного обучения.

Дистилляция - создание уменьшенной версии модели машинного обучения.

Дообучение (англ. fine-tuning) - обучение классификатора на основе уже существующей модели векторизации на определенном наборе данных.

Мессенджер - это система обмена мгновенными сообщениями между пользователями на компьютере или смартфоне.

В соответствии со схемой, представленной на фиг. 1, система обработки данных содержит устройство 1 клиента и устройство 10 виртуального помощника. Упомянутые устройства могут быть реализованы на базе по меньшей мере одного вычислительного устройства, выполненные в программно-аппаратной части таким образом, чтобы выполнять приписанные им ниже функции. Например, устройство 1 клиента может представлять собой портативный или стационарный компьютер, телефон, смартфон, планшет или прочее вычислительное устройство, выполненное с возможностью формирования и передачи текстовых или голосовых сообщений и данных для аутентификации клиента в устройство 10 виртуального помощника, а также запросов на предоставление по меньшей мере одного сервиса. Обмен данными между устройством 1 клиента и устройством 10 может осуществляться с использованием проводных или беспроводных каналов передачи данных, широко известных из уровня техники.

Устройство 10 виртуального помощника может включать: модуль 11 аутентификации, модуль 12 предоставления сервисов, модуль 13 токенизации текста, модуль 14 векторизации, модуль 15 классификации и модуль 16 извлечения сущностей. Модуль 12 предоставления сервисов и модуль 16 извлечения сущностей могут быть реализованы на базе программно-аппаратных средств устройства 10 виртуального помощника.

Модуль 11 аутентификации может быть реализован на базе вычислительного устройства и представлять функции личного кабинета. Личный кабинет - это особый раздел сайта, который позволяет клиенту определенной компании получить доступ к данным о состоянии и статистической информации лицевого счета, деталям заказа, ведущимся про проект работам и т.д. Для клиентов, являющихся представителями юридического лица, уполномоченное лицо юридического лица (ЮЛ), имеющее право на принятие решений от лица такого ЮЛ (далее - Уполномоченное лицо), широко известными из уровня техники методами авторизуется в модуле 11 аутентификации с помощью заранее созданной учетной записи,

принимает оферту и подключает свою компанию - ЮЛ к использованию сервисов, предоставляемых виртуальным ассистентом. После подключения по меньшей мере одного сервиса модуль 11 аутентификации сохраняет в памяти, которой он может быть оснащен, идентификатор (ID) ЮЛ, которому доступны сервисы, предоставляемые виртуальным ассистентом. В альтернативном варианте реализации представленного решения в памяти модуля 11 может быть задан список ЮЛ и доступных для них сервисов.

Далее Уполномоченное лицо может добавлять клиентов - представителей ЮЛ, например, работников компании, в список возможных пользователей сервисом, предоставляемых виртуальным ассистентом, путем указания их ID, например, номеров телефонов или e-mail адресов в списке пользователей без соответствующей учетной записи. Клиенты, добавленные в список пользователей без учетной записи, т.е. указанные в качестве представителя ЮЛ, могут регистрироваться в личном кабинете виртуального ассистента с помощью e-mail адреса и/или номера телефона, а виртуальной ассистент сможет определить принадлежность пользователя к ЮЛ.

Модуль 13 токенизации текста может быть реализован на базе вычислительного устройства, сконфигурированного в программно-аппаратной части для выполнения алгоритма Byte-Pair-Encoding [1]. Особенностью данного алгоритма токенизации является то, что полученные токены не всегда соответствуют словам, они могут представлять собой части слов (окончания, аффиксы, основы) или даже отдельные символы. Это позволяет использовать небольшой размер словаря, снизить ресурсоемкость всего технического решения и увеличить скорость его работы, и при этом эффективно обрабатывать запросы пользователей с новыми словами (формами слов) или опечатками, что позволяет отказаться от использования лемматизации и проверки орфографии. Пример токенизации пользовательского ввода "поставить задачу Андрею организовать планирование": "поставить", "задач", "##у", "Андре", "##ю", "организ", "##овать", "планирование". Решеткой в примере помечены токены, которые являются продолжением предыдущих токенов.

Модуль 14 векторизации может быть реализован на базе вычислительного устройства и содержать модель векторизации текста, заранее обученную на русскоязычных текстах путем дистилляции [2] предобученной языковой модели в значительно меньшую нейронную сеть. В процессе дистилляции меньшая модель (модель ученик) учится сопоставлять текстам в обучающей выборке вектора максимально похожие на вектора, сопоставляемые этим же текстам большей моделью (моделью учителем). При обучении используется функция потерь - средне-квадратичное отклонение (англ. mean-squared error):

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2.$$

В качестве модели учителя может быть использована русскоязычная или мультиязычная нейросетевая языковая модель (например, одна из использованных моделей - [3]). Языковая модель в общем смысле - это вероятностное распределение по последовательности слов (токенов). Нейросетевые языковые модели, как правило, состоят из двух основных частей: кодировщика (англ. encoder) и декодировщика (англ. decoder). Задача кодировщика сопоставить входной текстовой последовательности вектор или последовательность векторов. Задача декодировщика может варьироваться, но, как правило, состоит в том, чтобы, используя полученное из кодировщика векторное представление предсказать (т.е. приписать вероятность) следующему элементу текстовой последовательности или же в том, чтобы восстановить изначальную текстовую последовательность, используя лишь часть полученного из кодировщика векторного представления (англ. Masked Language Modelling). Такая постановка задачи позволяет в качестве обучающих данных использовать любой осмысленный текст на естественном языке, что в свою очередь позволяет получить векторные представления текстовых последовательностей очень высокого качества для широкого ряда задач. В последующем переиспользуется, как правило, только часть кодировщика. Обучение языковых моделей представляет собой очень ресурсоемкую задачу, однако многие полученные модели публикуются разработчиками в публичном доступе.

Переиспользование публично-доступных языковых моделей представляет большую ценность, так как позволяет получить качественный результат без затрат ресурсов на обучение. Однако недостатком таких моделей является их большой размер и, соответственно, трудности в эксплуатации. Этим обуславливается необходимость в дистилляции подобных моделей. Дистилляция осуществляется на данных, непосредственно представляющих целевой домен. Для рассматриваемого виртуального цифрового ассистента - это короткие тексты бизнес тематики, релевантные для физических и юридических лиц. Ограничение тематики текстов позволяет нивелировать неизбежное ухудшение модели вследствие снижения количества параметров - доступные параметры будут целенаправленно использованы только для представления целевой тематики и дистиллированная модель будет в целом хуже векторизовать тексты, однако будет сравнимо хорошо справляться с текстами целевой тематики.

При выборе архитектуры нейронной сети ученика предпочтение отдается рекуррентным сетям, в частности двунаправленным сетям долгой краткосрочной памяти (англ. BiLSTM - Bidirectional Long Short Term Memory [4]). При ограничениях на размеры они превосходят архитектуру Transformer, которая составляет основу предобученных языковых моделей.

Модуль 15 классификации может быть реализован на базе вычислительного устройства и содер-

жать модель классификации текста, обученной на основе дистиллированной модели, которая в свою очередь обучена на заранее созданной обучающей выборке. К модели векторизации (BiLSTM нейронной сети) добавляется полносвязный слой (англ. Fully-connected layer) с количеством ячеек равному количеству классов, к которому применяется функция активации softmax:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{k=1}^K e^{z_k}}$$

где z - вектор размерности K , полученный из модели; e - экспонента; i - индекс в z . Softmax позволяет получить из числового вектора вероятностное распределение по классам. При дообучении веса модели векторизации "замораживаются", т. е. они не меняются, подбираются только параметры последнего полносвязного слоя. При обучении используется функция потерь - кросс-энтропия (англ. cross-entropy или logloss). Для бинарной классификации она рассчитывается:

$$-(y \log(p) + (1 - y) \log(1 - p))$$

где y - правильная метка класса (0 или 1); \log - натуральный логарифм; p - предсказанная вероятность класса.

При количестве классов >2 она рассчитывается:

$$-\sum_{c=1}^M y_{o,c} \log(p_{o,c})$$

где M - количество классов, y - показатель правильности класса "с" для наблюдения "о" (0 или 1), p - предсказанная вероятность отнесения наблюдения "о" к классу "с".

Для того чтобы разделить бизнес-потребности физических и юридических клиентов создается два отдельных классификатора, каждый из которых обучается на размеченных данных соответствующей тематики, в частности на размеченной текстовой информации, содержащей запросы на предоставление сервисов. Помимо непосредственно векторизованного текстового ввода в классификаторы передается дополнительная мета-информация о пользователе, которая позволяет уточнить его статус в случаях, когда смысловой составляющей запроса недостаточно. Обучение классификаторов может осуществляться широко известными из уровня техники методами, например, раскрытыми в статье Васильев В. Г. "Обучение классификаторов на основе выделения фрагментов", Dialog, 2010, 11.05.2010 16:56:54, размещенной в Интернет по адресу: <http://www.dialog-21.ru/media/1486/11.pdf>.

Представленная на фиг. 1 система обработки данных работает следующим образом.

При первом обращении клиента к устройству 10 виртуального ассистента устройство 1 клиента перенаправляется в модуль 11 аутентификации для аутентификации клиента в упомянутом устройстве 10 известными из уровня техники методами. Например, клиенту может быть предложено ввести ID клиента на специализированном сайте или посредством установки специализированного приложения. Также при использовании телекоммуникационной связи, либо каналов связи мессенджеров для взаимодействия с виртуальным ассистентом клиент может быть идентифицирован по номеру телефона и/или учетной записи в мессенджере, ассоциированных с устройством 1 клиента.

После прохождения процесса аутентификации клиент посредством устройства 1 клиента может направить запрос на предоставления одного или более сервисов в устройство 10 виртуального ассистента. Например, клиент может запросить сервис по предоставлению статистики посещаемости, а запрос на предоставление услуги может представлять собой голосовое или текстовое сообщение "покажи статистику посещаемости с 12 по 16 декабря". Также виртуальный ассистент может быть выполнен с возможностью предоставления следующих сервисов:

- Отслеживание изменений законодательства;
- Проверка контрагента;
- Отслеживание отзывов о компании;
- Создание и работа с задачами в CRM-системах;
- Планирование и постановка встреч в календарь;
- Напоминания;
- Ведение протокола встречи;
- Управление складскими остатками (прием и выдача со склада);
- Отслеживание появления новых резюме необходимого профиля на онлайн-платформах;
- Сбор и классификация заявок в техподдержку;
- Заполнение данных по объектам.

Соответственно, направленный клиентом посредством устройства 1 запрос на предоставление сервиса поступает в модуль 12 предоставления сервисов, который известными из уровня техники методами преобразует полученный запрос в текстовую информацию и направляет полученную текстовую информацию в модуль 13 токенизации текста, который преобразует ее в последовательность токенов. Например, текстовая информация "покажи статистику посещаемости с 12 по 16 декабря" может быть представлена в виде последовательности токенов ['покажи', 'статистику', 'посещаем', '##ости', 'с', '12', 'по', '16', 'де-

кабря'].

Далее последовательность токенов направляется упомянутым модулем 13 в модуль 14 векторизации, который преобразует полученную последовательность токенов в численный вектор фиксированной размерности (например, 512 или 768), после чего упомянутый вектор направляется в модуль 15 классификации для его классификации. Классификация упомянутого вектора может выполняться посредством определения вероятностного распределения по известным классам, заранее сохраненным в упомянутом модуле 15, после чего модуль 15 выбирает класс с максимальным значением вероятности, если это значение соответствует значению заранее установленного порога. Таким образом, для текстовой информации "покажи статистику посещаемости с 12 по 16 декабря" будет определен класс, характерный для сервиса по предоставлению статистики посещаемости. Также поскольку модель, используемая для классификации, содержит два отдельных классификатора, каждый из которых обучается на размеченных данных, содержащих запросы на предоставление сервисов, относящихся как к клиентам - физическим лицам, так и представителям ЮЛ, то полученный класс будет относиться либо к клиентам - физическим лицам, либо представителям ЮЛ. Например, класс, характерный для сервиса по предоставлению статистики посещаемости, относится к представителям ЮЛ. Идентификатор (ID) выбранного класса упомянутым модулем 15 направляется в модуль 12 предоставления сервисов, который определяет, требуется ли клиенту быть представителем ЮЛ, например, работником компании. Список классов, для которых требуется быть представителем ЮЛ, может быть заранее задан в памяти упомянутого модуля 15. Если модулем 12 определено, что для данного класса быть представителем ЮЛ не требуется, то модуль 12 на основе класса определяет сервис, который следует оказать клиенту и предоставляет клиенту запрошенный им сервис, в частности, направляет статистику посещаемости в устройство 1 клиента.

Если модулем 12 предоставления сервисов определено, что для полученного класса клиента должен быть представителем ЮЛ, то упомянутый модуль 12 обращается к модулю 11 аутентификации для проверки, является ли клиент представителем ЮЛ. Если клиент не является представителем ЮЛ, то модуль 12 отказывает клиенту в предоставлении сервиса, направив в устройство 1 клиента соответствующее уведомление. Если упомянутым модулем 12 определено, что клиент является представителем ЮЛ, то модуль 12 на основе класса определяет сервис, который следует оказать клиенту и предоставляет клиенту запрошенный им сервис, в частности, направляет статистику посещаемости в устройство 1 клиента.

Таким образом, обеспечивается возможность определения того, что клиент является представителем юридического лица для предоставления ему соответствующего сервиса.

В альтернативном варианте реализации представленного решения модуль 12 предоставления сервисов после получения ID класса и определения того, что клиент должен быть представителем ЮЛ, может обратиться к упомянутому модулю 11 для определения ЮЛ, представителем которого клиент является и проверить, возможно ли предоставление сервиса данному ЮЛ в отношении данного класса. Если упомянутым модулем 12 определено, что для ЮЛ, представителем которого является клиент, предоставление запрошенного сервиса невозможно, то модуль 12 отказывает клиенту в предоставлении сервиса, направив в устройство 1 клиента соответствующее уведомление. Если модулем 12 определено, что для ЮЛ, представителем которого является клиент, предоставление запрошенного сервиса разрешено, то модулем 12 предоставляет клиенту сервис методом, описанным ранее.

Дополнительно модуль 12 предоставления сервисов может быть выполнен с возможностью уточнения параметров сервиса, который будет предоставлен клиенту, на основе атрибутов сущностей, содержащихся в текстовой информации. Для этого упомянутый модуль 12 направляет текстовую информацию в модуль 16 извлечения сущностей, который известными из уровня техники методами извлекает набор сущностей и их атрибуты и направляет в ответ в модуль 12. При получении набора сущностей и их атрибутов модуль 12 уточняет параметры сервиса, который будет предоставлен клиенту.

Например, для сервиса по предоставлению статистики посещаемости параметры сервиса могут характеризовать временной интервал, за который упомянутую статистику следует предоставить. Соответственно, если набор сущностей и их атрибуты характеризуют временной интервал, за который упомянутую статистику следует предоставить, то клиенту на устройство 1 клиента будет предоставлен уточненный сервис, т.е. статистика за указанный в текстовой информации временной интервал.

Также модуль 16 извлечения сущностей может быть использован для аугментации (обогащения) текстовых обучающих данных. Аугментации активно применяются в компьютерном зрении (например, к распространенным аугментациям относятся повороты, отражения, добавление шума). Аугментации в автоматической обработке языка практически не применяются, так как невозможно простым способом сгенерировать похожие по смыслу грамматически корректные тексты. Извлечение и нормализация сущностей с восстановлением или контекстным разрешением атрибутов позволяет аугментировать текстовые данные и улучшить качество классификации.

Аугментация данных осуществляется следующим образом. После того, как модуль 16 извлечения сущностей извлечет набор сущностей из текстовой информации, упомянутый модуль 16 на основе набора сущностей определяет контекст текстовой информации. Наборы сущностей и соответствующий им контекст текстовой информации может быть заранее задан разработчиком в памяти модуля 16. Например, для примера текстовой информации "покажи статистику посещаемости с 12 по 16 декабря" набор сущно-

стей указывает на наличие в текстовой информации даты начала временного периода с атрибутом сущности "12", даты окончания временного периода с атрибутом сущности "16" и месяца окончания временного периода с атрибутом сущности "декабрь", а контекст текстовой информации указывает на то, что текстовая информация характеризует временной период.

Далее модуль 16 извлечения сущностей сравнивает извлеченный набор сущностей с заданным набором сущностей для данного контекста текстовой информации для определения отсутствующих в текстовой информации сущностей. Набор сущностей для контекста текстовой информации может быть заранее задан разработчиком в памяти модуля 16. Например, для контекста текстовой информации, характеризующего временной период, заданный набор сущностей может содержать: дату начала временного периода, месяц начала временного периода, год начала временного периода, дату окончания временного периода, месяц окончания временного периода, год окончания временного периода. Соответственно, в приведенном примере "покажи статистику посещаемости с 12 по 16 декабря" модуль 16 определит, что в текстовой информации отсутствуют сущности, характеризующие месяц начала временного периода, год начала временного периода, год окончания временного периода.

После того, как модуль 16 извлечения сущностей определил отсутствующие сущности, упомянутый модуль 16 дополняет извлеченный набор сущностей отсутствующими сущностями на основе контекста текстовой информации. В частности, для указанного ранее примера набор сущностей, характеризующий наличие в текстовой информации даты начала временного периода с атрибутом сущности "12", даты окончания временного периода с атрибутом сущности "16" и месяца окончания временного периода с атрибутом сущности "декабрь", будет дополнен сущностями, характеризующими месяц начала временного периода, год начала временного периода, год окончания временного периода.

Что касается сущности, характеризующей год окончания временного периода, то она может быть определена модулем 16 следующим образом. На первом этапе модуль 16 определяет дату поступления запроса на предоставление сервиса посредством обращения к модулю 12, в частности, число, месяц и год поступления запроса. Далее на основе даты поступления запроса и месяца окончания временного периода модуль 16 определяет год окончания временного периода, после чего формирует сущность, характеризующую год окончания временного периода, которая далее может быть включена в извлеченный набор сущностей. Например, если запрос на предоставление сервиса поступил в январе 2021, а месяц окончания временного периода - "декабрь", то поскольку нужный интервал находится в прошлом, то год окончания временного периода модулем 16 будет определен как 2020 год. Соответственно, сформированная модулем 16 сущность будет характеризовать год окончания временного периода с атрибутом сущности "2020"

Что касается сущностей, характеризующей месяц начала временного периода, то она может быть определена модулем 16 на основе сущности, характеризующей дату начала временного периода и сущности, характеризующей дату окончания временного периода. Например, модуль 16 может сравнить сущность, характеризующую дату начала временного периода с атрибутом сущности "12", и сущность, характеризующую дату окончания временного периода с атрибутом сущности "16". Поскольку 12 меньше 16, следовательно, то в качестве месяца начала временного периода модулем 16 будет выбран месяце окончания временного периода "декабрь", после чего модуль 16 сформирует сущность, характеризующую месяц начала временного периода с атрибутом сущности "декабрь". Соответственно, если дата начала временного периода окажется больше даты окончания временного периода, то в качестве месяце окончания временного периода будет выбран "ноябрь".

Что касается сущностей, характеризующей год начала временного периода, то она может быть определена модулем 16 на основе сущностей, характеризующих месяц начала временного периода, месяц окончания временного периода и год окончания временного периода, например, путем сравнения месяца начала временного периода и месяца окончания временного периода. Поскольку в указанном ранее примере месяц начала временного периода "декабрь" совпадает с месяцем окончания временного периода "декабрь", то модуль 16 выбирает в качестве года начала временного периода - год окончания временного периода "2020", после чего модуль 16 сформирует сущность, характеризующую год начала временного периода с атрибутом сущности "2020". В то же время если месяц окончания временного периода - "январь", а месяц начала временного периода - "декабрь", то модуль 16 в качестве года начала временного периода выберет предыдущий год, т.е. 2019.

Таким образом, извлеченный набор сущностей после его дополнения недостающими сущностями будет представлять следующий вид:

```
{"from": {"day":12, "month": 12, "year":2020},
"to": {"day":16, "month": 12, "year":2020}}
```

где "day", "month", "year" - названия сущностей, а 12, 2020, 16 - атрибуты сущностей.

С помощью этого представления можно сгенерировать такие варианты, как "покажи статистику посещаемости с 12 декабря по 16 декабря", "покажи статистику посещаемости с 12 декабря по 16 декабря 2020", "покажи статистику посещаемости с 12 по 16 декабря 2020 года", "покажи статистику посещаемости с 12.12 по 16.12", "покажи статистику посещаемости 12-16 декабря". Также для аугментации кон-

кретные даты могут быть заменены на другие при условии, что получаются корректные периоды (даты существуют, начало интервала по времени стоит раньше конца). Таким образом, итоговыми аугментациями могут быть: "покажи статистику посещаемости с 4 ноября по 16 декабря", "покажи статистику посещаемости с 5 мая по 23 июня 2019", "покажи статистику посещаемости с 2.02 по 8.03", "покажи статистику посещаемости 1-7 августа", "покажи статистику посещаемости с 01.08.20 по 31.08.2020".

В общем виде (см. фиг. 2) вычислительное устройство (200) содержит объединенные общей шиной информационного обмена один или несколько процессоров (201), средства памяти, такие как ОЗУ (202) и ПЗУ (203) и интерфейсы ввода/вывода (204).

Процессор (201) (или несколько процессоров, многоядерный процессор и т.п.) может выбираться из ассортимента устройств, широко применяемых в настоящее время, например, таких производителей, как: Intel™, AMD™, Apple™, Samsung Exynos™, MediaTEK™, Qualcomm Snapdragon™ и т.п. Под процессором или одним из используемых процессоров в системе (200) также необходимо учитывать графический процессор, например, GPU NVIDIA с программной моделью, совместимой с CUDA, или Graphcore, тип которых также является пригодным для полного или частичного выполнения способа, а также может применяться для обучения и применения моделей машинного обучения в различных информационных системах.

ОЗУ (202) представляет собой оперативную память и предназначено для хранения исполняемых процессором (201) машиночитаемых инструкций для выполнения необходимых операций по логической обработке данных. ОЗУ (202), как правило, содержит исполняемые инструкции операционной системы и соответствующих программных компонент (приложения, программные модули и т.п.). При этом, в качестве ОЗУ (202) может выступать доступный объем памяти графической карты или графического процессора.

ПЗУ (203) представляет собой одно или более устройств постоянного хранения данных, например, жесткий диск (HDD), твердотельный накопитель данных (SSD), флэш-память (EEPROM, NAND и т.п.), оптические носители информации (CD-R/RW, DVD-R/RW, BlueRay Disc, MD) и др.

Для организации работы компонентов устройства (200) и организации работы внешних подключаемых устройств применяются различные виды интерфейсов В/В (204). Выбор соответствующих интерфейсов зависит от конкретного исполнения вычислительного устройства, которые могут представлять собой, не ограничиваясь: PCI, AGP, PS/2, IrDa, FireWire, LPT, COM, SATA, IDE, Lightning, USB (2.0, 3.0, 3.1, micro, mini, type C), TRS/Audio jack (2.5, 3.5, 6.35), HDMI, DVI, VGA, Display Port, RJ45, RS232 и т.п.

Для обеспечения взаимодействия пользователя с устройством (200) применяются различные средства (205) В/В информации, например, клавиатура, дисплей (монитор), сенсорный дисплей, тач-пад, джойстик, манипулятор мышь, световое перо, стилус, сенсорная панель, трекбол, динамики, микрофон, средства дополненной реальности, оптические сенсоры, планшет, световые индикаторы, проектор, камера, средства биометрической идентификации (сканер сетчатки глаза, сканер отпечатков пальцев, модуль распознавания голоса) и т.п. Средство сетевого взаимодействия (206) обеспечивает передачу данных посредством внутренней или внешней вычислительной сети, например, Интранет, Интернет, ЛВС и т.п. В качестве одного или более средств (206) может использоваться, но не ограничиваясь: Ethernet карта, GSM модем, GPRS модем, LTE модем, 5G модем, модуль спутниковой связи, NFC модуль, Bluetooth и/или BLE модуль, Wi-Fi модуль и др.

Конкретный выбор элементов устройства (200) для реализации различных программно-аппаратных архитектурных решений может варьироваться с сохранением обеспечиваемого требуемого функционала.

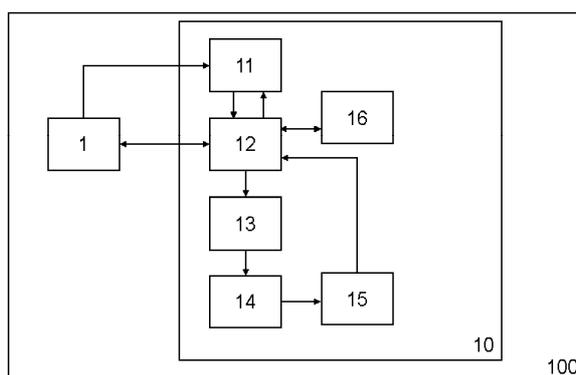
Модификации и улучшения вышеописанных вариантов осуществления настоящего технического решения будут ясны специалистам в данной области техники. Предшествующее описание представлено только в качестве примера и не несет никаких ограничений. Таким образом, объем настоящего технического решения ограничен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

Используемые источники информации

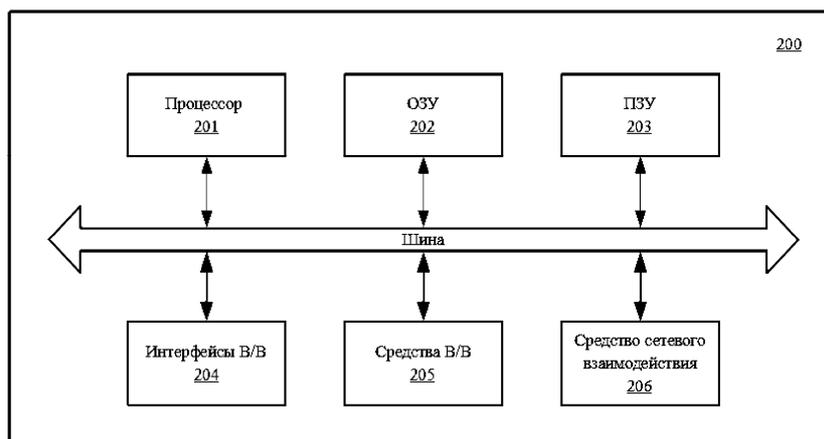
1. Sennrich, R., Haddow, B., & Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. ArXiv, abs/1508.07909.
2. Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the Knowledge in a Neural Network. arXiv preprint arXiv:1503.02531.
3. Feng, Fangxiaoyu, Yin-Fei Yang, Daniel Matthew Cer, N. Arivazhagan and Wei Wang. "Language-agnostic BERT Sentence Embedding." ArXiv abs/2007.01852 (2020): n. pag.
4. Greff, Klaus, R. Srivastava, J. Koutnik, Bastiaan Steunebrink and J. Schmidhuber. "LSTM: A Search Space Odyssey." IEEE Transactions on Neural Networks and Learning Systems 28 (2017): 2222-2232.

ФОРМУЛА ИЗОБРЕТЕНИЯ

1. Способ предоставления клиенту по меньшей мере одного сервиса, выполняемый по меньшей мере одним вычислительным устройством, содержащий этапы, на которых:
- получают запрос на предоставления по меньшей мере одного сервиса;
 - преобразуют полученный запрос в текстовую информацию;
 - классифицируют текстовую информацию посредством модели, содержащей два классификатора, причем один классификатор обучен на размеченных текстовых данных, содержащих запросы на предоставление сервисов, предоставляемых физическим лицам (ФЛ), а второй классификатор обучен на размеченных текстовых данных, содержащих запросы на предоставление сервисов, предоставляемых юридическим лицам (ЮЛ);
 - на основе полученного класса определяют, что для предоставления сервиса клиенту необходимо быть представителем ЮЛ;
 - определяют, что клиент является представителем ЮЛ;
 - на основе класса определяют сервис, который следует предоставить клиенту, и предоставляют упомянутый сервис клиенту.
2. Способ по п.1, характеризующийся тем, что дополнительно содержит этапы, на которых:
- преобразуют текстовую информацию в последовательность токенов;
 - преобразуют последовательность токенов в численный вектор фиксированной размерности.
3. Способ по п.1, характеризующийся тем, что дополнительно содержит этап, на котором определяют, что для ЮЛ, представителем которого является клиент, предоставление запрошенного сервиса разрешено.
4. Способ по п.1, характеризующийся тем, что дополнительно содержит этапы, на которых:
- извлекают набор сущностей и их атрибуты из текстовой информации;
 - уточняют параметры сервиса, который будет предоставлен клиенту, на основе извлеченного набора сущностей и их атрибутов.
5. Способ по п.4, характеризующийся тем, что дополнительно содержит этапы, на которых:
- на основе набора сущностей определяют контекст текстовой информации;
 - сравнивают набор сущностей с заданным набором сущностей для данного контекста текстовой информации для определения отсутствующих в текстовой информации сущностей;
 - дополняют извлеченный набор сущностей отсутствующими сущностями на основе контекста текстовой информации.
6. Устройство предоставления клиенту по меньшей мере одного сервиса, содержащее по меньшей мере одно вычислительное устройство и по меньшей мере одно устройство памяти, содержащее машиночитаемые инструкции, которые при их исполнении по меньшей мере одним вычислительным устройством выполняют способ по пп.1-5.



Фиг. 1



Фиг. 2

