(19) Евразийское патентное ведомство

(11) 040619 (13) B1

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ЕВРАЗИЙСКОМУ ПАТЕНТУ

(45) Дата публикации и выдачи патента
2022.07.06

(21) Номер заявки
202092855

(22) Дата подачи заявки
2020.12.23

(51) Int. Cl. ***G06F 40/10*** (2020.01)
G06F 40/279 (2020.01)
G06N 3/08 (2006.01)

(54) СИСТЕМА И СПОСОБ АУГМЕНТАЦИИ ОБУЧАЮЩЕЙ ВЫБОРКИ ДЛЯ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ

(31) **2020132305**
(32) **2020.04.28**
(33) **RU**
(43) **2021.10.29**

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

(72) Изобретатель:
Шаврина Татьяна Олеговна (RU)

(74) Представитель:
Герасин Б.В. (RU)

(56) CN-A-110795552
US-A1-20190325308
US-A1-20180329883
US-B2-9971763
CN-A-110796160

(57) Настоящее изобретение относится к области компьютерной техники, в частности к решениям для работы с алгоритмами машинного обучения в ходе формирования обучающих выборок. Технический результат заключается в повышении точности подбора текстовых данных на основании характеристик текста входной обучающей выборки. Заявленный результат достигается с помощью системы аугментации обучающей выборки для алгоритмов машинного обучения, которая содержит по меньшей мере один процессор; по меньшей мере одно средство памяти; модуль обработки входных данных, выполненный с возможностью получения текстовых данных, формирующих исходную обучающую выборку; и нормализации данных, при которой выполняется разделение текста на предложения и очистка текста от символов; модуль векторизации данных, выполненный с возможностью преобразования в векторную форму нормализованных предложений, при этом в ходе упомянутого преобразования осуществляется разбиение каждого полученного предложения на минимально значимые части, представляющие собой слова и знаки препинания; токенизация упомянутых минимально значимых частей; формирование векторных представлений для каждого токена; и формирование усредненного векторного представления нормализованного предложения; модуль обогащения текстовых данных, содержащий набор текстовых данных, собираемых из открытых источников, и метаданные, для их векторизации и построения поискового индекса; модуль текстового индекса, выполненный с возможностью формирования текстового индекса по векторным представлениям текстовых данных; модуль аугментации обучающей выборки, выполненный с возможностью дополнения и/или корректировки исходной текстовой выборки на основании подбора релевантных векторных представлений токенов в модуле обогащения текстовых данных с помощью определения меры близости токенов в векторном пространстве.

040619 B1

040619 B1

Область техники

Настоящее изобретение относится к области компьютерной техники, в частности к решениям для работы с алгоритмами машинного обучения в ходе формирования обучающих выборок.

Уровень техники

Под аугментацией данных может подразумеваться увеличение объема обучающей выборки в алгоритмах машинного обучения, причем увеличение объема может быть как искусственное, произведенное за счет видоизменения имеющейся выборки, так и за счет фильтрации подходящих открытых ресурсов с опорой на имеющуюся выборку. В настоящий момент задача аугментации текстовых данных требуется в широком ряде направлений и отраслей, связанных с машинным обучением. В частности, в построении диалоговых систем (чат-боты, умные помощники) применение аугментации данных делает системы более устойчивыми к вариативности команд и естественной синонимии в речи. В промышленных областях, где требуется классификация документов, но собственных текстовых данных в отрасли накоплено мало (или они недоступны для разработчиков из-за своей закрытости - это медицинские данные, юридические документы, государственная документация), также прибегают к аугментации данных, чтобы улучшить качество работы классификации в условиях реального применения.

Также одним из направлений, нуждающихся в аугментированных данных, является извлечение информации (извлечение именованных сущностей и связей между ними). Огромная вариативность имен персоналий, названий компаний и локаций требует от обучающей выборки большого объема и разнообразных контекстов, в которых сущности употребляются. Открытые данные в этом направлении покрывают лишь малую часть возможных случаев употребления сущностей и не являются достаточными для промышленной реализации таких систем.

В настоящее время используется ряд подходов, каждый из которых обладает своими преимуществами и недостатками. Случайные перестановки слов в данных, случайные удаления слов, замены слов на синонимы и морфологические аналоги.

Известен способ аугментации данных (https://arxiv.org/abs/1901.11196 EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks), который применяется для задач, где реализуется анализ последовательностей и классификация, но при этом часть данных становится трудночитаемой и менее понятной для пользователя, а также не воспринимается носителями языка как корректное, понятное высказывание.

Известно также применение онтологического/семантического подхода. Некоторые слова в данных меняются на более общие/частные понятия, что помогает системам делать более общие и более точные частные выводы, однако приносит пользу только в задачах, где не требуется устойчивость относительно формулировок предложения/команд/порядка слов/стиля высказывания. Достаточно небольшое количество слов языка попадает в структурированную онтологию.

Автоматический перевод. Используются открытые системы перевода с языка на язык (Google Translate). Данные переводятся на несколько популярных языков, затем происходит обратный перевод на исходный язык. Подход дает наиболее полное перефразирование исходных данных, однако достаточно часто меняет смысл исходных высказываний настолько далеко, что увеличивает зашумленность исходных данных. Таким образом, существенным недостатком известных подходов является отсутствие возможности дополнения/корректировки обучающих выборок с сохранением релевантности данных по отношению к входной информации для целей исключения потери смысловой составляющей текста.

Сущность изобретения

Решением существующей технической проблемы в данной области техники является создание системы аугментации данных на основании анализа распределения данных с помощью формирования глобального текстового индекса, дополняемого из открытых источников данных.

Технический результат заключается в обеспечении подбора текстовых данных для аугментации обучающей выборки на основании характеристик текста входной обучающей выборки.

Заявленный результат достигается с помощью системы аугментации обучающей выборки для алгоритмов машинного обучения, которая содержит

по меньшей мере один процессор;

по меньшей мере одно средство памяти;

модуль обработки входных данных, выполненный с возможностью

получения текстовых данных, формирующих исходную обучающую выборку; и

нормализации данных, при которой выполняется разделение текста на предложения и очистка текста от символов;

модуль векторизации данных, выполненный с возможностью преобразования в векторную форму нормализованных предложений, при этом в ходе упомянутого преобразования осуществляется

разбиение каждого полученного предложения на минимально значимые части, представляющие собой слова и знаки препинания;

токенизация упомянутых минимально значимых частей;

формирование векторных представлений для каждого токена; и

формирование усредненного векторного представления нормализованного предложения;

модуль обогащения текстовых данных, содержащий набор текстовых данных, собираемых из открытых источников, и метаданные, для их векторизации и построения поискового индекса;

модуль текстового индекса, выполненный с возможностью формирования текстового индекса по векторным представлениям текстовых данных;

модуль аугментации обучающей выборки, выполненный с возможностью дополнения и/или корректировки исходной текстовой выборки на основании подбора релевантных векторных представлений токенов в модуле обогащения текстовых данных с помощью определения меры близости токенов в векторном пространстве.

В одном из частных примеров реализации системы модуль векторизации данных формирует усредненное векторное представление текста.

В другом частном примере реализации системы размерность усредненного векторного представления равна 768:1.

В другом частном примере реализации системы метаданные включают в себя по меньшей мере одно из следующего: ссылка на источник в глобальной сети Интернет, дата источника, жанр, дата создания, данные автора, рубрика, тематика, количество слов в источнике.

В другом частном примере реализации системы мера близости токенов и текстов в пространстве представляет собой косинусную меру близости.

В другом частном примере реализации системы в векторном пространстве каждый токен имеет уникальные координаты.

В другом частном примере реализации системы на основании координат определяются минимальные и максимальные граничные значения пространства текстов исходной обучающей выборки.

В другом частном примере реализации системы аугментация обучающей выборки осуществляется с помощью добавления новых текстов, имеющих координаты, не выходящие за пределы граничных значений.

В другом частном примере реализации системы дополнение исходной обучающей выборки осуществляется до заданного пользователем количества слов. В другом частном примере реализации системы осуществляется итеративный поиск ближайших текстов в векторном пространстве для каждого текста из предложений исходной выборки.

В другом частном примере реализации системы уникальность подбираемых текстов определяется на основании метаданных, хранимых в модуле обогащения текстовых данных.

Заявленное решение также осуществляется с помощью компьютерно-реализуемого способа аугментации обучающей выборки для алгоритмов машинного обучения, при этом способ выполняется с помощью по меньшей мере одного процессора и содержит этапы, на которых

получают текстовые данные исходной обучающей выборки;

выполняют нормализацию данных, при которой выполняется разделение текста на предложения и очистка текста от символов;

выполняют векторизацию нормализованных предложений, при этом в ходе упомянутого преобразования осуществляется

разбиение каждого полученного предложения на минимально значимые части, представляющие собой слова и знаки препинания (токенизация); и

формирование векторных представлений для каждого нормализованного текста на основании входящих в него токенов (значимых частей);

формируют текстовый индекс по векторным представлениям текстовых данных, при этом текстовый индекс формируется из векторного пространства, формируемого из текстов, расположенных в открытых источниках, и метаданных;

осуществляют аугментацию исходной обучающей выборки с помощью подбора релевантных векторных представлений текстов на основании определения меры близости в векторном пространстве на основании поискового индекса.

В одном из частных примеров осуществления способа при векторизации текстовых данных формируется усредненное векторное представление текста.

В другом частном примере осуществления способа размерность усредненного векторного представления равна 768:1.

В другом частном примере осуществления способа метаданные включают в себя по меньшей мере одно из следующего: ссылка на источник в глобальной сети Интернет, дата источника, жанр, дата создания, данные автора, рубрика, тематика, количество слов в источнике.

В другом частном примере осуществления способа мера близости токенов и текстов в пространстве представляет собой косинусную меру близости.

В другом частном примере осуществления способа в векторном пространстве каждый токен имеет уникальные координаты.

В другом частном примере осуществления способа на основании координат определяются минимальные и максимальные граничные значения пространства текстов исходной обучающей выборки.

В другом частном примере осуществления способа аугментация обучающей выборки осуществля-

ется с помощью добавления новых текстов, имеющих координаты, не выходящие за пределы граничных значений.

В другом частном примере осуществления способа дополнение исходной обучающей выборки осуществляется до заданного пользователем количества слов.

В другом частном примере осуществления способа осуществляется итеративный поиск ближайших текстов в векторном пространстве для каждого текста из предложений исходной выборки.

В другом частном примере осуществления способа уникальность подбираемых текстов определяется на основании метаданных.

Краткое описание чертежей

Фиг. 1 иллюстрирует пример заявленной системы.

Фиг. 2 иллюстрирует блок-схему заявленного способа.

Фиг. 3 иллюстрирует общий вид вычислительного устройства.

Осуществление изобретения

Заявленное решение осуществляется с помощью компьютерной системы (100), представленной на фиг. 1, которая может выполняться на базе компьютерного устройства, например персонального компьютера, сервера и т.п. Система аугментации обучающих выборок включает в себя основные функциональные элементы, такие как модуль обработки входных данных (101), модуль векторизации (102), модуль обогащения данных (103), модуль текстового индекса (104) и модуль аугментации (105). Модуль обработки входных данных (101) включает в себя предобработку текстов пользователя, передаваемых в систему аугментации. Также модуль (101) осуществляет их чистку и преобразование в общее пространство численных признаков. Входные текстовые данные разделяются на предложения. Существующие открытые технологии позволяют провести данную операцию для русского языка без дополнительной разработки. Входной формат текстовой выборки, как правило, представляет собой .txt. Деление полученного текста на предложения осуществляется с помощью открытых библиотек на языке python3 (например, https://pypi.org/project/rusenttokenize/). Также с помощью модуля (101) выполняется деление предложений входной выборки на токены с помощью разбиения предложений по пробелам и отделения от них знаков препинания.

На выходе модуля обработки входных данных (101) формируется список предложений и токенов в них.

Пример

"Все люди смертны. Сократ - человек. Следовательно, Сократ смертен." → ["Все люди смертны.", "Сократ - человек.", "Следовательно, Сократ смертен."].

Далее модуль (101) осуществляет очистку текстов от спецсимволов. Так как для векторного пространства необходимо представлять текст как точку в многомерном пространстве признаков слов (векторных представлений), то спецсимволы, не относящиеся к буквам, цифрам и знаками препинания, способны внести в этот вектор шум и сместить положение текста в пространстве признаков относительно других, что критично для итогового качества подбора и корректировки текстовой выборки в ходе аугментации.

С помощью обработки входной информации модулем (101) происходит фильтрация входящих предложений от спецсимволов, не входящих в список кириллических и латинских букв, чисел и символов со стандартной 105-клавишной клавиатуры. Такая очистка позволяет очистить текст от шумов, которые внесут неизвестные универсальной модели редкие символы, и сделать полученные векторы более точными. Фильтрация происходит при помощи регулярных выражений.

Пример.

"•_Мама_ мыла раму.😊" → "Мама мыла раму.".

На выходе работы модуля (101) получается список предложений входной обучающей выборки, очищенных от спецсимволов.

Модуль векторизации (102) представляет собой одну или несколько моделей машинного обучения для преобразования текстовой информации в векторную форму - эмбеддинг. Векторизации подлежат очищенные предложения текста, полученные с помощью модуля (101). Могут применяться модели машинного обучения, основанные на пословной векторизации или получении вектора всего контекста предложения целиком. В модуле векторизации (102) предпочтительно применять модели машинного обучения, например искусственные нейронные сети (ИНС), которые способны делать генерализированный вывод о мире, обученные на большом объеме закрытых данных (тексты на десятки миллиардов слов - обычно корпуса новостей, блогов, литературы, в том числе технической, открытых энциклопедий), для обработки и анализа свойств новых текстов. Такие модели, как BERT, ELMo, ULMFit, XLNet, RoBerta и др. уже успешно применяются для русского языка в задачах обработки малых данных. С помощью использования одного или нескольких из вышеуказанных решений модулем (102) может осуществляться формирование векторных представлений текстов и предложений в эмбеддинги. Эмбеддинги, полученные на основании универсальной модели, обладающей генерализованными знаниями о вариативности текстов, позволяют оценить их положение в многомерном пространстве свойств текстов вообще и дополнить вы-

борку текстами, по своим численным признакам похожими на исходные тексты обучающей выборки выборки пользователя.

В качестве примера можно рассмотреть применение модели BERT для русского языка (http://docs.deeppavlov.ai/en/master/features/pretrained_vectors.html). Модель выступает в качестве источника получения эмбеддингов предложения. Модуль векторизации (102) на основании полученных от модуля (101) нормализованных текстовых данных входной обучающей выборки осуществляет разбиение каждого предложения на минимально значимые части - токены (слова, знаки препинания).

Токенизация (разделение текста на токены) происходит с помощью открытой технологии, подходящей для модели BERT, например BertTokenizer (см. https://pypi.org/project/pytorch-pretrained-bert/). По итогам токенизации формируется список строк, соответствующих токенам предложения. Для каждого токена с помощью модуля векторизации (102) передается эмбеддинг, который берется из последнего - 11-го слоя модели BERT. Эмбеддинг имеет размерность 768 на 1. Для каждого предложения формируется соответствующий эмбеддинг заданной размерности (в данном решении используется размерность вектора 1×768) с помощью нейросетевой модели. В частности, данный эмбеддинг может формироваться с помощью операции усреднения токенов.

Модуль обогащения данных (103) представляет собой базу данных с текстами из открытых источников данных, например веб-ресурсов с различными вариантами текстов, литературы и т.п. Модуль (103) содержит тексты суммарным объемом 10 млрд слов, при этом выполнен с возможностью постоянного наполнения, что обеспечивает большую вариативность контекстов материала на русском языке с учетом различных стилей, жанров и типов материалов.

Информация, содержащаяся в модуле обогащения (103), служит исходным материалом, корпусом текстов для создания полноценного индекса натуральных текстов, материалами которого будет дополняться передаваемая выборка. Помимо самих текстов, в модуле (103) хранятся доступные метаданные о тексте, такие как

идентификатор (ID);

информация об источнике, адрес его местонахождения в сети Интернет (url, ip-адрес и т.п.);

дата добавления в хранилище;

жанр;

дата написания;

ФИО автора;

рубрика, тематика;

количество слов.

Модуль текстового индекса (104) обеспечивает формирование иерархического индекса на базе предварительно векторизованных текстов из модуля (103). Векторизация текстовых данных в модуле (103) осуществляется с помощью модуля векторизации (102). Построение индекса производится при помощи библиотеки (https://pypi.org/project/nmslib/).

Данная библиотека имеет методы индексирования, максимально подходящие для построения индекса на эмбеддингах: можно построить иерархический индекс, подбирающий максимально похожие тексты на основании косинусной меры. Данная мера близости является популярной метрикой, используемой для получения языковых объектов (слов, предложений, текстов), максимально схожих по своим свойствам, закодированным в эмбеддингах.

Косинусная мера близости определяется с помощью скалярного произведения и нормы между двумя векторами по формуле

$$Similarity(AB) = Cos(\theta) = \frac{A * B}{\| A \| * \| B \|}$$

Широкая применимость косинусной меры, в частности, в задачах информационного поиска, машинного обучения и обработки текста обусловлена ее эффективностью в качестве оценочной меры для разреженных векторов/эмбеддингов, так как необходимо учитывать только ненулевые значения эмбеддингов (а таких нулевых значений в текстовых эмбеддингах бывает достаточно, так как это означает, что какой-то признак в тексте отсутствует).

Косинусная мера является лишь частным примером способа индексирования; он может быть любым. В данном случае уместно использовать иерархический индекс по причине того, что он достаточно компактен и при этом обеспечивает быстрое получение ближайших объектов по эмбеддингам. Потенциально возможно использовать и любые другие методы построения индекса на косинусной мере (sparse cosine similarity indexing), но из-за немалой размерности эмбеддингов (обычно они включают последовательности от 300 до 2000 чисел, в заявленном решении - 768) иерархические методы осуществляют наиболее быстрый поиск самого близкого объекта в индексе к объекту запроса. В рамках эксперимента был собран тестовый индекс, построенный на 100000 случайных предложений из русской Википедии и вебкорпуса Common Crawl (блоги, новости, реклама). Были собраны заголовки новостей и популярных записей в блогах и к ним подобраны максимально похожие предложения из тестового индекса: на приведенных ниже примерах можно наблюдать, как в подобранных предложениях сохраняется тематика, эмоциональная окраска предложений, стиль и лексические признаки.

Для полного индекса создается индекс на данных из открытого веб-корпуса Omnia Russica объемом 33 млрд слов на русском языке (собран автором данной заявки): https://omnia-russica.github.io/.

Модуль аугментации выборки (105) представляет собой набор моделей для определения полноты выборки, полученной модулем (101). Для последующей аугментации исходной обучающей выборки модуль (105) может функционировать в двух режимах работы:

1) создание скорректированной и/или дополненной выборки;

2) дополнение выборки до требуемого количества слов.

Увеличение выборки до требуемого объема осуществляется на основании пользовательского ввода, который указывает желаемый объем выборки в словах, что позволяет достичь максимально достижимого на данном индексе значения; например, если пользователь хочет 1 млрд слов, а есть только 20 млн, выдается 20 млн. На фиг. 2 представлена блок-схема выполнения способа аугментации обучающей выборки (200). На первом этапе (201) пользователь загружает в систему исходную текстовую выборку, которая обрабатывается модулем (101) и в последующем преобразовывается в векторное представление (202).

В случае корректировки выборки происходит следующая операция. По полученным векторам входной выборки (тексты, полученные от пользователя для аугментации) вычисляются экстремальные значения по каждой переменной эмбеддингов - минимум и максимум - по каждой из 768 переменных в эмбеддинге. Полученные 768 минимумов и максимумов образуют гиперпространство в пространстве признаков модели аугментации, применяемой модулем (105).

Из сформированного иерархического индекса (203) извлекаются все тексты, эмбеддинги которых попадают в упомянутое гиперпростанство, т.е. эмбеддинги, удовлетворяющие условиям минимума и максимума по каждой переменной в координатном пространстве. Список подобных примеров предложений выводится в текстовом виде. Аугментация выборки (204) в части улучшения (корректировки) выборки достигается за счет обогащения ее новыми примерами, которые не выделяются экстремальными значениями, при этом позволяют получить более точное понимание распределения интересующих пользователя явлений, например, на основании совпадений тематики текстов, частоты упоминания терминов в векторном пространстве и т.п. Аугментация выборки (204) в части ее дополнения до требуемого количества слов осуществляется следующим образом. По полученным векторам токенов входной выборки (тексты, полученные от пользователя для аугментации) вычисляются экстремальные значения по каждой переменной эмбеддингов аналогично способом, упомянутым выше для улучшения выборки, которые формируют векторное гиперпространство текстовых данных.

Из сформированного текстового индекса (203) извлекаются все тексты, чьи эмбеддинги попадают в данное гиперпростанство, что позволяет оценить объем полученной текстовой выборки.

Если объем текстовой выборки меньше заявленного пользователем количества слов, то происходит следующая операция: по индексу подбираются по N (начиная с N=1) максимально близких по косинусной мере предложений к каждому предложению из полученной выборки, даже если они не входят в определенное гиперпространство. Итеративно увеличивая число N на единицу, осуществляется циклический перебор всех предложений для поиска уникальных похожих текстов до тех пор, пока количество слов не достигнет установленного пользователем числа. Уникальность примеров контролируется проверкой по id предложения в базе модуля (103).

Если выборка меньше заявленного пользователем количества слов, то происходит следующая операция: все примеры, полученные из гиперпространства признаков, сортируются по схожести на основании вычисления косинусной меры близости к примерам в пользовательской выборке. Происходит циклический перебор каждого примера из пользовательской выборки и для него подбирается N наиболее близких примеров. Параметр N итеративно увеличивается на 1, пока количество слов в полученной выборке не составит заявленное число.

Выполнение способа аугментации выборки (200) позволяет подобрать наиболее релевантные текстовые данные, существующие в постоянно формируемом пространстве иерархического текстового индекса, которые применяются для обогащения входной обучающей выборки пользователя.

Заявленное решение возможно встраивать в другие системы для улучшения их работы, например систему автоматической разметки сущностей в тексте (задача named entity recognition - под сущностями имеются в виду персоны, локации, названия организаций, иногда дополнительные сущности; задача является сложной, так как для ее решения требуется подбор большого количества размеченных примеров). При работе в составе системы для разметки сущностей пользователь загружает неразмеченные данные и примеры сущностей, затем данные искусственно аугментируются по вышеописанному способу (200), разметка сущностей происходит с учетом большего количества контекстов, которые формируются при аугментации выборки.

Сама по себе идея поиска дополнительных данных часто осуществляется вручную на ограниченном наборе открытых источников. Однако такой подход абсолютно не учитывает вариативность в исходных текстовых данных, так как текст все же следует рассматривать математически как последовательность редких событий с большим количеством факторов, влияющих на распределение - стиль, жанр, источник, цель и дата написания, отношение автора с адресатом и т.д. Добавление неоднородных текстовых данных к исходной выборке способно полностью нивелировать ее особенности и ухудшить результаты обу-

чения. С помощью реализации заявленного подхода процесс поиска подходящей дополняющей однородной выборки автоматизируется, при этом происходит учет вариативности особенностей текста.

На фиг. 3 представлен общий вид вычислительного устройства (300). На базе устройства (300) может быть реализовано устройство пользователя для формирования и загрузки выборки, вычислительное устройство (100) для выполнения способа аугментации (200) и иные непредставленные устройства, которые могут участвовать в общей информационной архитектуре заявленного решения.

В общем случае вычислительное устройство (300) содержит объединенные общей шиной информационного обмена один или несколько процессоров (301), средства памяти, такие как ОЗУ (302) и ПЗУ (303), интерфейсы ввода/вывода (304), устройства ввода/вывода (305) и устройство для сетевого взаимодействия (306).

Процессор (301) (или несколько процессоров, многоядерный процессор) могут выбираться из ассортимента устройств, широко применяемых в текущее время, например, компаний Intel™, AMD™, Apple™, Samsung Exynos™, MediaTEK™, Qualcomm Snapdragon™ и т.п. Процессор (301) может включать в себя также графический процессор или работать в совокупности с графическим ускорителем, например Nvidia, AMD Radeon и др., которые могут применяться для осуществления вычислительных операций при выполнении алгоритмов машинного обучения.

ОЗУ (302) представляет собой оперативную память и предназначено для хранения исполняемых процессором (301) машиночитаемых инструкций для выполнение необходимых операций по логической обработке данных. ОЗУ (302), как правило, содержит исполняемые инструкции операционной системы и соответствующих программных компонент (приложения, программные модули и т.п.).

ПЗУ (303) 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 (300) и организации работы внешних подключаемых устройств применяются различные виды интерфейсов В/В (304).

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 вычислительным устройством (300) применяются различные средства (305) В/В информации, например клавиатура, дисплей (монитор), сенсорный дисплей, тачпад, 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 (306) обеспечивает передачу данных устройством (300) посредством внутренней или внешней вычислительной сети, например Интранет, Интернет, ЛВС и т.п. В качестве одного или более средств (306) может использоваться, но не ограничиваться, 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.

Дополнительно могут применяться также средства спутниковой навигации в составе устройства (300), например GPS, ГЛОНАСС, BeiDou, Galileo.

Представленные материалы заявки раскрывают предпочтительные примеры реализации технического решения и не должны трактоваться как ограничивающие иные, частные примеры его воплощения, не выходящие за пределы испрашиваемой правовой охраны, которые являются очевидными для специалистов соответствующей области техники.

ФОРМУЛА ИЗОБРЕТЕНИЯ

1. Система аугментации обучающей выборки для алгоритмов машинного обучения, содержащая

по меньшей мере один процессор;

по меньшей мере одно средство памяти;

модуль обработки входных данных, выполненный с возможностью

получения текстовых данных, формирующих исходную обучающую выборку; и

нормализации данных, при которой выполняется разделение текста на предложения и очистка текста от символов;

модуль векторизации данных, выполненный с возможностью преобразования в векторную форму нормализованных предложений, при этом в ходе упомянутого преобразования осуществляется

разбиение каждого полученного предложения на минимально значимые части, представляющие собой слова и знаки препинания;

токенизация упомянутых минимально значимых частей;

формирование векторных представлений для каждого токена; и

формирование усредненного векторного представления нормализованного предложения;

модуль обогащения текстовых данных, содержащий набор текстовых данных, собираемых из открытых источников, и метаданные, для их векторизации и построения поискового индекса;

модуль текстового индекса, выполненный с возможностью формирования текстового индекса по векторным представлениям текстовых данных;

модуль аугментации обучающей выборки, выполненный с возможностью дополнения и/или корректировки исходной текстовой выборки на основании подбора релевантных векторных представлений токенов в модуле обогащения текстовых данных с помощью определения меры близости токенов в векторном пространстве.

2. Система по п.1, характеризующаяся тем, что модуль векторизации данных формирует усредненное векторное представление текста.

3. Система по п.2, характеризующаяся тем, что размерность усредненного векторного представления равна 768:1.

4. Система по п.1, характеризующаяся тем, что метаданные включают в себя по меньшей мере одно из следующего: ссылка на источник в глобальной сети Интернет, дата источника, жанр, дата создания, данные автора, рубрика, тематика, количество слов в источнике.

5. Система по п.1, характеризующаяся тем, что мера близости токенов и текстов в пространстве представляет собой косинусную меру близости.

6. Система по п.1, характеризующаяся тем, что в векторном пространстве каждый токен имеет уникальные координаты.

7. Система по п.6, характеризующаяся тем, что на основании координат определяются минимальные и максимальные граничные значения пространства текстов исходной обучающей выборки.

8. Система по п.7, характеризующаяся тем, что аугментация обучающей выборки осуществляется с помощью добавления новых текстов, имеющих координаты, не выходящие за пределы граничных значений.

9. Система по п.8, характеризующаяся тем, что дополнение исходной обучающей выборки осуществляется до заданного пользователем количества слов.

10. Система по п.9, характеризующаяся тем, что осуществляется итеративный поиск ближайших текстов в векторном пространстве для каждого текста из предложений исходной выборки.

11. Система по п.10, характеризующаяся тем, что уникальность подбираемых текстов определяется на основании метаданных, хранимых в модуле обогащения текстовых данных.

12. Компьютерно-реализуемый способ аугментации обучающей выборки для алгоритмов машинного обучения, выполняемый с помощью по меньшей мере одного процессора и содержащий этапы, на которых

получают текстовые данные исходной обучающей выборки;

выполняют нормализацию данных, при которой выполняется разделение текста на предложения и очистка текста от символов;

выполняют векторизацию нормализованных предложений, при этом в ходе упомянутого преобразования осуществляется

разбиение каждого полученного предложения на минимально значимые части, представляющие собой слова и знаки препинания (токенизация); и

формирование векторных представлений для каждого нормализованного текста на основании входящих в него токенов (значимых частей);

формируют текстовый индекс по векторным представлениям текстовых данных, при этом текстовый индекс формируется из векторного пространства, формируемого из текстов, расположенных в открытых источниках, и метаданных;

осуществляют аугментацию исходной обучающей выборки с помощью подбора релевантных векторных представлений текстов на основании определения меры близости в векторном пространстве на основании поискового индекса.

13. Способ по п.12, характеризующийся тем, что при векторизации текстовых данных формируется усредненное векторное представление текста.

14. Способ по п.13, характеризующийся тем, что размерность усредненного векторного представления равна 768:1.

15. Способ по п.12, характеризующийся тем, что метаданные включают в себя по меньшей мере одно из следующего: ссылка на источник в глобальной сети Интернет, дата источника, жанр, дата создания, данные автора, рубрика, тематика, количество слов в источнике.

16. Способ по п.12, характеризующийся тем, что мера близости токенов и текстов в пространстве представляет собой косинусную меру близости.

17. Способ по п.12, характеризующийся тем, что в векторном пространстве каждый токен имеет уникальные координаты.

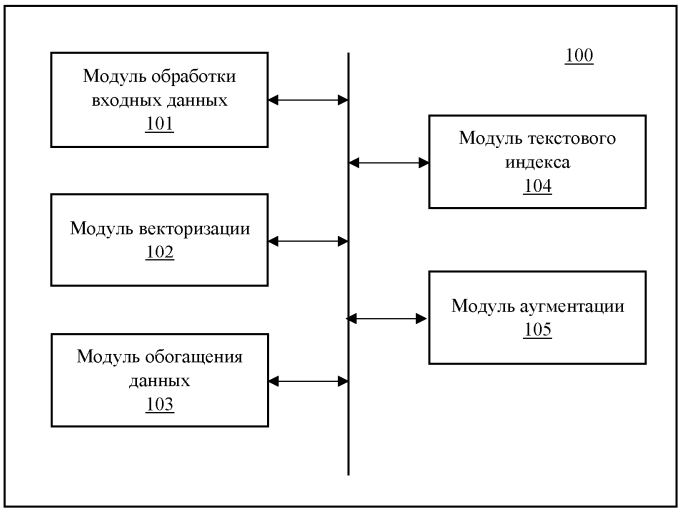
18. Способ по п.17, характеризующийся тем, что на основании координат определяются минимальные и максимальные граничные значения пространства текстов исходной обучающей выборки.

19. Способ по п.18, характеризующийся тем, что аугментация обучающей выборки осуществляется с помощью добавления новых текстов, имеющих координаты, не выходящие за пределы граничных значений.

20. Способ по п.19, характеризующийся тем, что дополнение исходной обучающей выборки осуществляется до заданного пользователем количества слов.

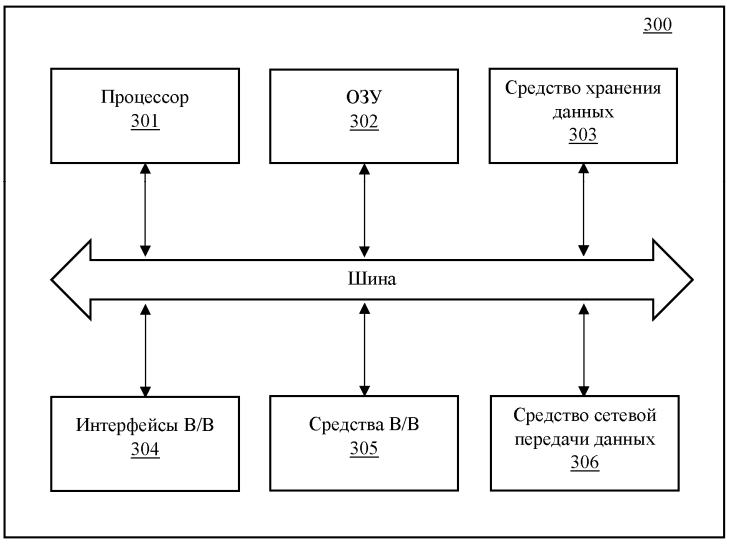
21. Способ по п.20, характеризующийся тем, что осуществляется итеративный поиск ближайших текстов в векторном пространстве для каждого текста из предложений исходной выборки.

22. Способ по п.21, характеризующийся тем, что уникальность подбираемых текстов определяется на основании метаданных.



Фиг. 1

Фиг. 2


300
Процессор
301
ОЗУ
302
Средство хранения данных
303
Шина
Интерфейсы В/В
304
Средства В/В
305
Средство сетевой передачи данных
306

Фиг. 3