(19) Евразийское патентное ведомство

(11) 039466 (13) B1

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ЕВРАЗИЙСКОМУ ПАТЕНТУ

(45) Дата публикации и выдачи патента
2022.01.31

(21) Номер заявки
201992491

(22) Дата подачи заявки
2019.11.18

(51) Int. Cl. ***G06F 21/60*** (2006.01)
G06F 40/279 (2006.01)
G06N 20/00 (2006.01)

(54) СПОСОБ И СИСТЕМА КЛАССИФИКАЦИИ ДАННЫХ ДЛЯ ВЫЯВЛЕНИЯ КОНФИДЕНЦИАЛЬНОЙ ИНФОРМАЦИИ В ТЕКСТЕ

(31) **2019132817**
(32) **2019.10.16**
(33) **RU**
(43) **2021.04.30**

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

(72) Изобретатель:
Теренин Алексей Алексеевич, Котова Маргарита Александровна (RU)

(74) Представитель:
Герасин Б.В. (RU)

(56) US-A1-20180365593
US-B1-10412102
US-A1-20180367575
RU-C1-2661533

(57) Настоящее техническое решение, в общем, относится к области вычислительной обработки данных, а в частности, к методам классификации данных для выявления конфиденциальной информации. Компьютерно-реализуемый способ классификации данных для выявления конфиденциальной информации, выполняемый с помощью по меньшей мере одного процессора и содержащий этапы, на которых: получают данные, представленные в текстовом формате; осуществляют обработку полученных данных с помощью алгоритмов машинного обучения, в ходе которой каждому слову в тексте присваивается тег, соответствующий заданному типу конфиденциальной информации, причем для каждого алгоритма машинного обучения сформирована матрица классификации, на основании которой вычисляется F-мера для каждого типа данных; выполняют классификацию каждого слова в тексте на основе полученных от каждого алгоритма машинного обучения текстов с проставленными тегами и соответствующей алгоритмам машинного обучения матрицы F-мер и формируют итоговый вариант текста с проставленными тегами; выполняют классификацию текста с проставленными у каждого слова тегами по классам конфиденциальности на основе сравнения совокупности имеющихся тегов в тексте с заданными тегами конфиденциальной информации.

039466 B1

039466 B1

Область техники

Настоящее техническое решение, в общем, относится к области вычислительной обработки данных, а в частности, к методам классификации данных для выявления конфиденциальной информации.

Уровень техники

В настоящее время выявление конфиденциальной информации из большого массива данных и последующая ее классификация является приоритетной задачей для многих отраслей. Наиболее широкое применение данных технологий наблюдается в финансовом секторе, где среди больших объемов различных данных необходимо отдельно выявлять и классифицировать конфиденциальную информацию. Для этого используются различные инструменты и технологии, позволяющие так или иначе выявлять конфиденциальную информацию из больших объемов общих данных. Ключевой особенностью в работе таких инструментов является анализ данных с помощью алгоритмов машинного обучения.

Данные хранятся и обрабатываются в различных автоматизированных системах и файловых ресурсах, имеющих различные уровни конфиденциальности, способы доступа, атрибутивный состав. Проверка на наличие чувствительных данных осуществляется различными инструментами. В связи с этим появилась необходимость создать единое техническое решение, позволяющее с помощью нейронных сетей автоматически обрабатывать большое количество данных и выявлять конфиденциальную информацию. Для этого необходимо обучить искусственный интеллект распознавать содержимое документов, в которых может содержаться конфиденциальная информация. На сегодняшний момент из уровня техники известны решения, направленные на хранение и классификацию данных по заданным пользователем критериям.

Сервис Amazon Macie - сервис, который проводит мониторинг данных, использующий несколько методов автоматической классификации контента, чтобы идентифицировать и расставить приоритеты для конфиденциальных данных и точно определить ценность данных для бизнеса. Сервис распознаёт такую информацию, как персональная информация или интеллектуальная собственность. Одним из методов классификации является классификация по регулярному выражению. Классификация объектов с помощью регулярных выражений основана на конкретных данных или шаблонах данных, которые ищет Amazon Macie при проверке содержимого объектов данных. Amazon Macie предлагает набор управляемых регулярных выражений, каждый из которых имеет определенный уровень риска от 1 до 10. Также Amazon Macie классифицирует объекты с помощью метода опорных векторов.

Недостатками данного решения являются отсутствие возможности изменять существующие или добавлять новые регулярные выражения, возможность только включить или отключить поиск любых существующих регулярных выражений, сервис идентифицирует только те объекты, которые подходят под правила. Недостатки использования регулярных выражений заключаются в том, что для каждого вида конфиденциальной информации необходимо прописывать несколько регулярных выражений, которые не учитывают редкие особенности данных или могут быть более общими, например содержать в себе лишние данные.

Известно решение Google Cloud DLP, обеспечивающее быструю, масштабируемую классификацию и редактирование для чувствительных данных, таких как номера кредитных карт, имена, номера социального страхования, выбранные международные идентификаторы, номера телефонов и учетные данные GCP. Облако DLP классифицирует эти данные, используя более 90 предопределенных детекторов, чтобы идентифицировать шаблоны, форматы и контрольные суммы.

Недостаток данного решения заключается в использовании только регулярных выражений, для каждого вида конфиденциальной информации необходимо прописывать несколько регулярных выражений, которые не учитывают редкие особенности данных или могут быть более общими, например содержать в себе лишние данные.

Сущность технического решения

Заявленное техническое решение предлагает новый подход в области выявления и классификации конфиденциальной информации с помощью создания моделей машинного обучения для обработки большого объема данных.

Решаемой технической проблемой или технической задачей является создание нового способа классификации данных, обладающего высокой степенью точности и высокой скоростью распознавания конфиденциальной информации.

Основным техническим результатом, достигающимся при решении вышеуказанной технической проблемы, является повышение точности классификации конфиденциальной информации.

Дополнительным техническим результатом, достигающимся при решении вышеуказанной технической проблемы, является повышение скорости классификации конфиденциальной информации.

Заявленные результаты достигаются за счет компьютерно-реализуемого способа классификации данных для выявления конфиденциальной информации, выполняемого с помощью по меньшей мере одного процессора и содержащего этапы, на которых:

получают данные, представленные в текстовом формате;

осуществляют обработку полученных данных с помощью алгоритмов машинного обучения, в ходе которой каждому слову в тексте присваивается тег, соответствующий заданному типу конфиденциаль-

ной информации, причем для каждого алгоритма машинного обучения сформирована матрица классификации, на основании которой вычисляется F-мера для каждого типа данных;

выполняют классификацию каждого слова в тексте на основе полученных от каждого алгоритма машинного обучения текстов с проставленными тегами и соответствующей алгоритмам машинного обучения матрицы F-мер и формируют итоговый вариант текста с проставленными тегами;

выполняют классификацию текста с проставленными у каждого слова тегами по классам конфиденциальности на основе сравнения совокупности имеющихся тегов в тексте с заданными тегами конфиденциальной информации.

В одном из частных вариантов осуществления способа для каждого алгоритма машинного обучения вычисляются показатели F-меры для каждого типа данных.

В другом частном варианте осуществления способа конфиденциальная информация представлена, по меньшей мере, в виде текстовых данных и/или числовых данных.

Также указанные технические результаты достигаются за счет осуществления системы классификации данных для выявления конфиденциальной информации, которая содержит по меньшей мере один процессор; по меньшей мере одну память, соединенную с процессором, которая содержит машиночитаемые инструкции, которые при их выполнении по меньшей мере одним процессором обеспечивают выполнение вышеуказанного способа.

Описание чертежей

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

фиг. 1 иллюстрирует блок-схему выполнения заявленного способа;

фиг. 2 иллюстрирует пример извлекаемых именованных сущностей;

фиг. 3 иллюстрирует пример архитектуры CRF с Bi-LSTM;

фиг. 4 иллюстрирует пример размеченных данных для обучения моделей;

фиг. 5 иллюстрирует результаты обучения моделей;

фиг. 6 иллюстрирует результаты обучения моделей;

фиг. 7 иллюстрирует пример результатов проверки на тестовой выборке;

фиг. 8 иллюстрирует общий вид заявленной системы.

Осуществление изобретения

В данном техническом решении могут использоваться для ясности понимания работы термины и сокращения, которые будут расшифрованы далее в настоящих материалах заявки.

Модель в машинном обучении - совокупность методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи, а обучение в процессе применения решений множества сходных задач.

AI (Artificial Intelligence) - искусственный интеллект.

Токен - элемент последовательности из букв или слов или знак препинания.

Тег - значение, присваиваемое токену.

Задача теггирования последовательности (sequence labeling problem) - присвоение каждому элементу последовательности (токену) соответствующего тега.

Именованная сущность - это слово или словосочетание, обозначающее предмет или явление определенной категории.

Named entity recognition, NER - извлечение именованных сущностей - выделение из текста объектов, совокупности объектов и присвоение этим объектам категории, определяющей значение этих объектов (например, ФИО, названия организаций, локации).

RNN - сокращение от Recurrent neural network, рекуррентные нейронные сети.

LSTM - сокращение от Long ShortTerm Memory, долгая краткосрочная память - архитектура рекуррентной нейронной сети.

Bi-LSTM - сокращение от Bidirectional Long ShortTerm Memory, Двунаправленная Долгая краткосрочная память - архитектура рекуррентной нейронной сети.

NLP - сокращение от Natural Language Processing, обработка естественного языка.

CRF - сокращение от Conditional Random Field, условные случайные поля.

Word embeddings - векторное представление слов - сопоставление подаваемых на вход модели объектов векторам.

Заявленный способ (100) классификации данных для выявления конфиденциальной информации, как представлено на фиг. 1, заключается в выполнении ряда последовательных этапов, осуществляемых процессором вычислительного устройства.

Начальным шагом (101) является получение массива данных в текстовом формате. Текстовые данные содержат информацию, которая может представлять собой номера банковских карт, СНИЛС, ОКПО, ОГРН, ИНН, дату, номер паспорта, номер телефона, фамилию, имя, отчество, электронную почту, адрес, должность, адрес сайта и др., не ограничиваясь.

Следующим шагом (102) осуществляют обработку полученных данных с помощью алгоритмов машинного обучения, в ходе которой, каждому слову в тексте присваивается тег, соответствующий задан-

ному типу конфиденциальной информации, причем для каждой нейронной сети сформирована матрица классификации, на основании которой вычисляется F-мера для каждого типа данных.

Для того, чтобы модели могли обрабатывать данные, подаваемые им на вход, необходимо текст представить в форме, понятной, нейронным сетям. Для этого необходимо сопоставить все подаваемые на вход объекты векторам. В заявленном способе для этого используется комбинация векторного представления слов и символов. Комбинация методов вводится для улучшения качества работы модели. Буквы каждого слова в предложении подаются в Bi-LSTM сеть, для того чтобы выявить характеристики слов на символьном уровне. Отдельно создаётся векторное представление слов (token embedding). Затем векторное представление слов и символов конкатенируются, а затем подаются в модель Bi-LSTM+CRF. Стандартным компонентом нейронной сети для решения задач обработки естественного языка являются предобученные векторные представления слов.

Обучение нейронных сетей происходит на заранее размеченных данных. Каждому токену в последовательности ставится в соответствие тег (короткая строка, которая взаимно однозначно соответствуют видам конфиденциальной информации) из предварительно определенного набора тегов. Теги подбираются таким образом, чтобы пользователь мог интуитивно понять, что этот тег обозначает, например, CARD - номер карты, NAME - имя и т.д. Теги пишутся на латинице, для того, чтобы они имели общий вид на всех кодировках. Виды конфиденциальной информации входят в одну из категорий законодательно регулируемых данных, например персональные данные, банковская тайна, коммерческая тайна и т.д.

Для задачи NER есть несколько общих типов сущностей, которые по сути являются тегами. Для определения конфиденциальности документа необходимо умение извлекать следующие сущности: ФИО, дата, должность, почтовый индекс и т.д., не ограничиваясь. Пример извлекаемых именованных сущностей приведен на фиг. 2. Исходный текст токенизируется и тегируется. Для каждого токена есть отдельный тег с разметкой. Теги отделяются от токенов пробелами. Предложения разделены пустыми строками. Набор данных представляет собой текстовый файл или набор текстовых файлов. Набор данных должен быть разбит на три раздела: тренировочные, тестовые и валидационные. Тренировочные используются для обучения сети, а именно для регулировки весов с градиентным спуском. Валидация используется для мониторинга прогресса обучения и более ранней остановки.

Способ обучения нейронных сетей будет раскрыт далее в настоящих материалах заявки.

Матрица классификации - стандартный инструмент для оценки статистических моделей, в ней отображены вероятности распознавания действительного значения как прогнозируемого для каждого заданного прогнозируемого варианта.

На основе классификации тестовых данных вычисляются F-меры. F-мера или (F1-score) представляет собой совместную оценку точности и полноты. Данная метрика вычисляется по следующей формуле: F-мера = 2 * Точность * Полнота / (Точность + Полнота). F-мера вычисляется в каждом алгоритме для каждого вида данных.

Следующим шагом (103) выполняют классификацию каждого слова в тексте на основе полученных от каждой нейронной сети текстов с проставленными тегами и соответствующей нейронным сетям матрицы F-мер и формируют итоговый вариант текста с проставленными тегами.

Набор тегов заранее предопределен. Данный набор формируется для обозначение всех категорий данных, которые необходимо проверить на отнесение к конфиденциальным согласно законодательным, регуляторным, внутренним или иным нормам. Предобработка (векторизация) данных направлена на ускорение и облегчение дальнейшей обработки, а комбинации нескольких нейронных сетей - для повышения точности простановки тегов. Проставляется тот тег, на который с наибольшей вероятностью указывает не менее одной модели. Также модели должны проводить контекстный последовательный анализ, так как адреса, ФИО и некоторые другие типы данных в общем случае состоят из нескольких слов. Для подготовки текста к классификации он токенизируется и тегируется. Для каждого токена есть отдельный тег с разметкой. Теги отделяются от токенов пробелами. Предложения разделены пустыми строками. Итоговый набор данных представляет собой тегированный текстовый файл или набор текстовых файлов.

Для извлечения именованных сущностей в данном решении используется Yargy-парсер. Парсер для разметки - это готовый механизм, который способен извлекать имена, даты, локации, организации и т.д. Для улучшения работы парсера используется существующая библиотека, получившая имя "Natasha". Часть готовых правил парсинга уже доступны в библиотеке "Natasha". Для текущего решения правила для извлечения сущностей описываются с помощью контекстно-свободных грамматик и словарей, построенных на основе требований, заданных нормативными документами. Например, если существует несколько уровней критичности информации, то перечисляется, какие сущности должны относится к каждому из уровней.

В результате производится разбивка по предложениям, каждому предложению присваивается соответствующий номер. Сопоставляются все имеющиеся символы со словами в предложениях. Теги приводятся к категориальному типу.

Первый слой модели (Embedding слой) превращает последовательности чисел (слова сопоставили числам) в плотные векторы фиксированного размера. Далее используется оболочка TimeDistributed, что-

бы применить слой Embedding к каждой последовательности символов и получить векторное представление слов. Далее векторные представления слов и символов конкатенируются.

Полученные векторные представления подаются в основной слой модели (Bidirectional). Данный слой рассчитывает вероятности тегов для каждого слова в предложении. Далее эти вероятности подаются в слой CRF, который рассчитывает распределение вероятностей перехода от одного тега к другому.

На шаге (104) выполняют классификацию текста с проставленными у каждого слова тегами по классам конфиденциальности на основе сравнения совокупности имеющихся тегов в тексте с заданными тегами конфиденциальной информации.

Подготовленные на шаге (103) данные, используются для анализа и присвоения уровня классификации всего текста. Для этого используются подготовленные таблицы сочетания тегов, определяющих суммарный уровень конфиденциальности всего текста. Уровень конфиденциальности зависит от сочетания тегов, а не только от их наличия.

Пример: просто адрес во всем тексте не представляет критичности, а адрес с упоминанием обращающихся денежных средств уже должен классифицироваться с повышенной строгостью. Также, имя классифицирует целую группу людей и не может считаться критичным. Но имя с номером телефона уже персональные данные, которые должны классифицироваться на соответствующем уровне критичности. Если в документе содержится только ФИО, это один уровень, но если кроме ФИО имеется номер телефона и дата рождения, то уровень конфиденциальности документа намного выше. Для корректной классификации важно отслеживать контекст тегируемого словат, то есть оценивать сущности, находящиеся слева и справа. Теги получают приставки: "B-", если это первое вхождение тега данного типа, и "I-", если продолжение. Пример: "апрель [B-Date] 2019 [I-Date] года [I-Date]" или "117 [B-Money] миллионов [I-Money]".

Рекуррентные нейронные сети (RNN) используются для решения различных задач, включая проблемы обработки естественного языка из-за их способности использовать предыдущую информацию из последовательности для расчета текущего выхода.

Чтобы правильно обработать текущее слово в тексте (присвоить тег), необходимо, чтобы сеть основывалась на понимании предыдущего контекста. Значит она должна помнить, какой был текст слева от текущего слова. Традиционные нейронные сети не обладают этим свойством, их нельзя обучить долговременным зависимостям. Рекуррентные нейронные сети помогают решить данную проблему. Они содержат обратные связи, благодаря которым могут передавать информацию от одного шага сети к другому.

Однако, несмотря на возможность обучения долговременным зависимостям, на практике модели RNN не работают должным образом и страдают из-за проблемы исчезающего градиента. Данная проблема возникает по причине того, что сигналы об обратно распространяемых ошибках быстро становятся очень маленькими (или наоборот, чрезмерно большими). На практике они уменьшаются экспоненциально с количеством слоев в сети. По этой причине была разработана специальная архитектура RNN под названием долгая краткосрочная память (Long ShortTerm Memory - LSTM), чтобы справиться с исчезающим градиентом. Один повторяющийся модуль LSTM-сети состоит из четырёх слоев. LSTM заменяет скрытые блоки в архитектуре RNN на блоки, называемые блоками памяти, которые содержат четыре компонента: три вида фильтров (входной фильтр, фильтр забывания, выходной фильтр и ячейку памяти (memory cell). Правильное распознавание именованного объекта в предложении зависит от контекста. Предшествующие и последующие слова имеют значение для предсказания тега. Двунаправленные рекуррентные нейронные сети были разработаны для кодирования каждого элемента в последовательность с учетом левого и правого контекстов, что делает их одним из лучших выборов для задачи NER. Двунаправленная модель расчета состоит из двух этапов: прямой слой вычисляет представление левого контекста, обратный слой вычисляет представление правого контекста. Выходы этих шагов объединяются для получения полного представления элемента входной последовательности. Условные случайные поля (Conditional Random Field, CRF) - это ненаправленная вероятностная графическая модель для структурированного предсказания условных вероятностей событий, соответствующих вершинам некоторого графа, при условии наблюдаемых данных. Архитектура CRF с Bi-LSTM, применяемая для реализации способа (100), представлена на фиг. 3.

В комбинированной модели векторные представления слов (способ получения векторного представления описан в практической части) подаются в двунаправленную нейросеть Bi-LSTM. Эта сеть рассчитывает вероятности тегов для каждого слова в предложении. Пусть для входной последовательности слов (предложения) $X=(x_0, x_1, .., x_n)$ P - матрица вероятностей, которую выдаёт сеть Bi-LSTM. Эта матрица размером n*k, где k - число различных тегов, а n - длина входящей последовательности. $P_{i,j}$ - это вероятность, что у i-го слова в предложении тег j. Для последовательности ответов $y = \{y_0, y_1, .., y_N\}$ score вычисляется по следующей формуле:

$$\boldsymbol{Score}(\boldsymbol{X}, \boldsymbol{y}) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i,y_i}$$

где $A_{yi,yi+1}$ - обозначает вероятность, которая представляет собой оценку перехода от тега i к тегу j, то есть того, что на позиции j=i+1 будет именно тег y_{i+1}, при условии, что предыдущий тег y_i.

Для повышения точности предсказания слой CRF обучен обеспечивать соблюдение ограничений в зависимости от порядка тегов. Например, в схеме I O B (I - Внутри, O - Другое, B - Начало) тег I никогда не появляется в начале предложения, или O IB O - недопустимая последовательность тегов. Полный набор параметров для этой модели состоит из параметров Bi-LSTM слоев (весовые матрицы, смещения, матрица векторных представлений слов) и матрицы перехода CRF слоя. Все эти параметры настраиваются во время тренировки алгоритма обратного распространения ошибки со стохастическим градиентным спуском.

Далее будет представлен принцип обучения нейронных сетей и оценка качества моделей для целей осуществления заявленного способа.

На фиг. 4 представлен пример размеченных данных для обучения моделей.

Для обучения моделей необходимо подготовить размеченный датасет с текстом и тегами (фиг. 4). Производится разбивка по предложениям, каждому предложению присваивается соответствующий номер.

Пример: в подготавливаемом тексте 47959 предложений, содержащих 35179 различных слов. Получается 847657 строк в датасете.

На первом этапе обучения производится разбивка по предложениям, каждому предложению присваивается соответствующий номер. На следующем этапе приводят токены (каждое значение столбца Word на фиг. 4) к векторному виду. Для этого вводятся словари для слов, для символов и для тегов. Слова в предложениях сопоставляются с последовательностью чисел, а затем применяется к числовым последовательностям функцию pad_sequensec(), чтобы привести последовательности к одному размеру. Далее сопоставляются все имеющиеся символы со словами в предложениях, а теги приводятся к категориальному типу.

На следующем этапе разделяется выборка на тренировочную, валидационную и тестовую. Используется пропорции 80% к 10% к 10%.

Пример: тренировочная - 38846 предложений, валидационная - 4317 предложений, тестовая - 4796 предложений.

Следующим этапом обучения идет построение модели. Первым слоем является Embedding слой, задача которого: перевести последовательности чисел (которым сопоставили слова размеченного текста) в плотные векторы фиксированного размера. Таким образом получаем векторное представление слов.

Далее используется оболочка TimeDistributed, чтобы применить слой Embedding к каждой последовательности символов. После приведенной обработки получается векторное представление символов. Далее производится конкатенация векторных представлений слов и символов.

На следующем этапе задействуется основной слой модели - Bidirectional. Полученные на предыдущем этапе векторные представления подаются в слой Bidirectional. Данный слой рассчитывает вероятности тегов для каждого слова в предложении. Далее эти вероятности подаются в слой CRF, который рассчитывает распределение вероятностей перехода от одного тега к другому. Все параметры модели (весовые матрицы, смещения, матрица векторных представлений слов и матрица перехода CRF слоя) настраиваются во время тренировки алгоритма обратного распространения ошибки со стохастическим градиентным спуском.

Далее проводится тренировка/обучение модели.

На фиг. 5, 6 представлены результаты обучения Bi-LSTM+CRF модели. Здесь показывается рост точности обучения (accuracy) и рост точности на валидационных данных (validation accuracy) с ростом числа эпох, а также показано как уменьшались потери.

На фиг. 7 представлены результаты проверки на тестовой выборке. Сопоставляется текст, который был размечен человеком с результатами разметки, полученными на выходе модели. Качество работы модели определяется тем, насколько близко модель предугадала значение тега или, иными словами, насколько меньше отклонений у результата работы модели от предразмеченных значений тегов. Для оценки качества работы той или иной модели применяются общепринятые методы и метрики оценки качества моделей.

На фиг. 8 представлен пример общего вида вычислительной системы (300), которая обеспечивает реализацию заявленного способа (100) или является частью компьютерной системы, например сервером, персональным компьютером, частью вычислительного кластера, обрабатывающего необходимые данные для осуществления заявленного технического решения.

В общем случае система (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 (1105) и устройство для сетевого взаимодействия (306).

Процессор (301) (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 (300) также необходимо учитывать графический процессор, например GPU NVIDIA или Graphcore, тип которых также является пригодным для полного или частичного выполнения способа (100), а также может применяться для обучения и применения мо-

делей машинного обучения в различных информационных системах.

ОЗУ (302) представляет собой оперативную память и предназначено для хранения исполняемых процессором (301) машиночитаемых инструкций для выполнения 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.). При этом, в качестве ОЗУ (302) может выступать доступный объем памяти графической карты или графического процессора.

ПЗУ (303) представляет собой одно или более устройств постоянного хранения данных, например жесткий диск (HDD), твердотельный накопитель данных (SSD), флэш-память (EEPROM, NAKD и т.п.), оптические носители информации (CD-R/RW, DVD-R/RW, BlueRay Disc, MD) и др.

Для организации работы компонентов системы (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 вычислительной системой (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 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. 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. Компьютерно-реализуемый способ классификации данных для выявления конфиденциальной информации, выполняемый с помощью по меньшей мере одного процессора и содержащий этапы, на которых:

получают данные представленные в текстовом формате;

осуществляют обработку полученных данных с помощью алгоритмов машинного обучения, в ходе которой, каждому слову в тексте присваивается тег, соответствующий заданному типу конфиденциальной информации, причем для каждого алгоритма машинного обучения сформирована матрица классификации, на основании которой вычисляется F-мера для каждого типа данных;

выполняют классификацию каждого слова в тексте на основе полученных от каждого алгоритма машинного обучения текстов с проставленными тегами и соответствующей алгоритмам машинного обучения матрицы F-мер и формируют итоговый вариант текста с проставленными тегами;

выполняют классификацию текста с проставленными у каждого слова тегами по классам конфиденциальности на основе сравнения совокупности имеющихся тегов в тексте с заданными тегами конфиденциальной информации.

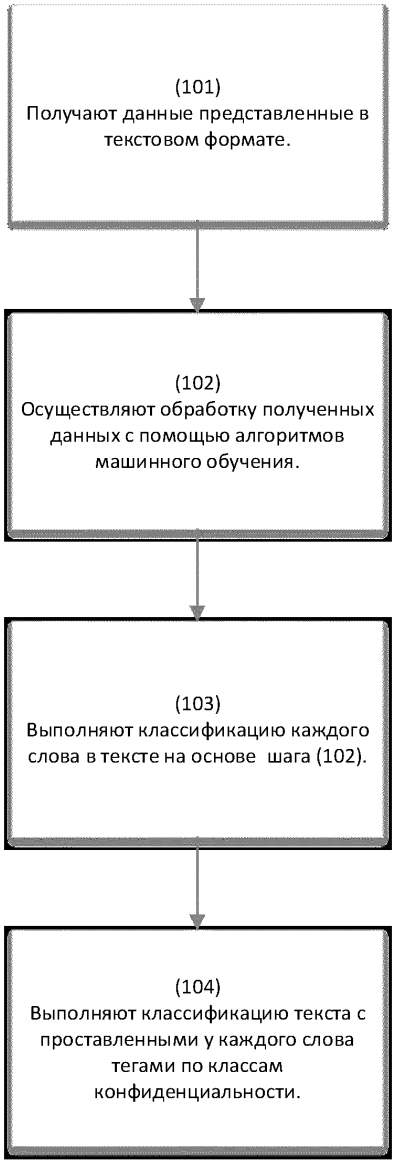
2. Способ по п.1, характеризующийся тем, что для каждого алгоритма машинного обучения вычисляются показатели F-меры для каждого типа данных.

3. Способ по п.1, характеризующийся тем, что конфиденциальная информация представлена, по меньшей мере, в виде текстовых данных и/или числовых данных.

4. Система классификации данных для выявления конфиденциальной информации, содержащая

по меньшей мере один процессор;

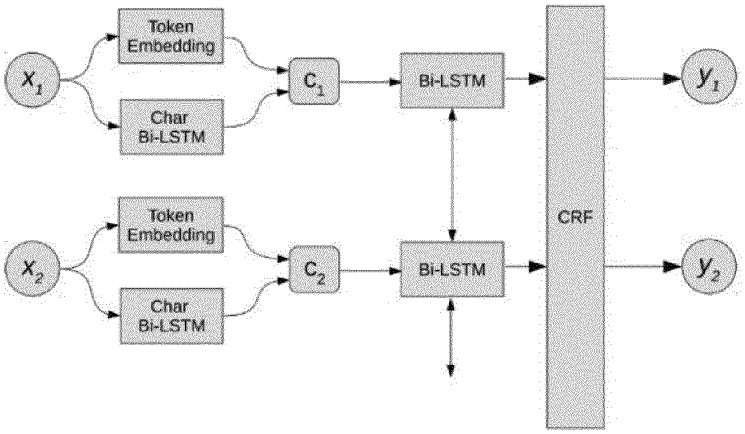
по меньшей мере одну память, соединенную с процессором, которая содержит машиночитаемые инструкции, которые при их выполнении по меньшей мере одним процессором обеспечивают выполнение способа по любому из пп.1-4.


(101)
Получают данные представленные в текстовом формате.
(102)
Осуществляют обработку полученных данных с помощью алгоритмов машинного обучения.
(103)
Выполняют классификацию каждого слова в тексте на основе шага (102).
(104)
Выполняют классификацию текста с проставленными у каждого слова тегами по классам конфиденциальности.

Фиг. 1

Именованная сущность
Адрес ЮЛ, ФЛ
Алиасы и доменные имена (кроме относительного пути)
Информация Департамента Безопасности
Информация Министерства по чрезвычайным ситуациям
Информация о VIP, МВК, премьер-клиентах
Данные системы СКУД
Местонахождение денег
Местонахождение ключей
Информация о защищённости ОСБ и банкоматов
Информация о маршрутах инкассации
Информация Управления внутрибанковский безопасности
Информация Федеральной службы безопасности
Информация Федеральной службы охраны
Информация Службы внешней разведки
Информация Национальной гвардии
Информация Министерства внутренних дел
Информация Министерства обороны
Информация Министерства юстиции
Основной номер держателя карты (PAN)
CAV2 CVC2 CVV2 CID
Дата
Должность ФЛ
ФИО
Дробное число
Отчество
Почтовый индекс
Номер ИНН ФЛ ЮЛ

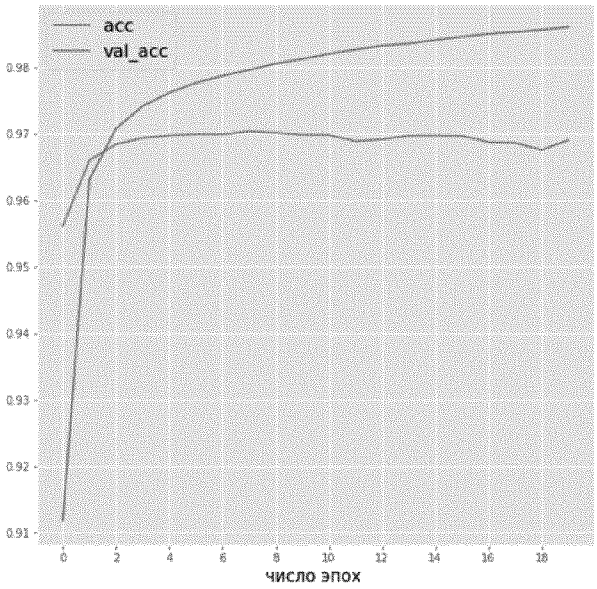
Фиг. 2



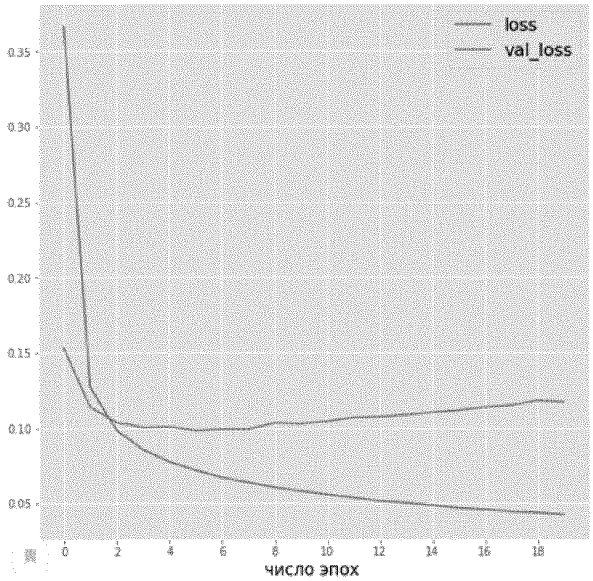
Фиг. 3

	Sentence: #	Word	Tag
0	1.0	25	B-DATE
1	1.0	апреля	I-DATE
2	1.0	Владимир	B-PERSON
3	1.0	Петров	I-PERSON
4	1.0	выступил	O
5	1.0	на	O
6	1.0	открытии	O
7	1.0	компании	O
8	1.0	"Северный	B-ORG
9	1.0	край"	I-ORG
10	1.0	.	O
11	2.0	МОСКВА	B-GEO
12	2.0	,	O
13	2.0	21	B-DATE
14	2.0	июня	I-DATE

Фиг. 4



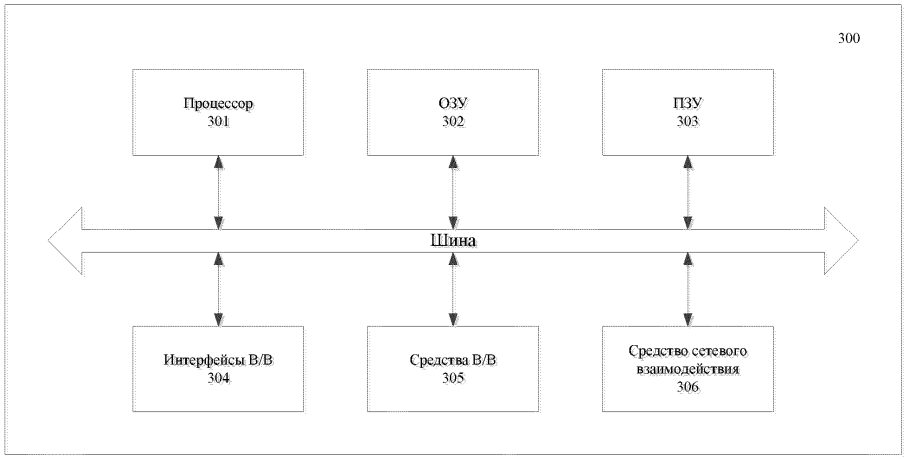
Фиг. 5



Фиг. 6

Word	True	Pred
Выручка	O	O
НЛМК	B-ORG	B-ORG
в	O	O
2011	B-DATE	B-DATE
году	I-DATE	I-DATE
составила	O	O
117	B-MONEY	B-MONEY
млрд	I-MONEY	I-MONEY
долларов	I-MONEY	I-MONEY
,	O	O
рентабельность	O	O
по	O	O
показателю	O	O
EBITDA	O	O
составила	O	O
195	B-PERCENT	B-PERCENT
%	I-PERCENT	I-PERCENT
.	O	O

Фиг. 7



Фиг. 8