

(19)



**Евразийское  
патентное  
ведомство**

(21) **202192993** (13) **A1**

(12) **ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ЕВРАЗИЙСКОЙ ЗАЯВКЕ**

(43) Дата публикации заявки  
**2023.03.31**

(51) Int. Cl. **G06F 16/35** (2006.01)  
**G06F 40/284** (2006.01)  
**G06F 40/30** (2006.01)

(22) Дата подачи заявки  
**2021.11.30**

(54) **СПОСОБ И СИСТЕМА ПОЛУЧЕНИЯ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ  
ЭЛЕКТРОННОГО ТЕКСТОВОГО ДОКУМЕНТА ДЛЯ КЛАССИФИКАЦИИ ПО  
КАТЕГОРИЯМ КОНФИДЕНЦИАЛЬНОЙ ИНФОРМАЦИИ**

(31) **2021128049**

(72) Изобретатель:

(32) **2021.09.24**

**Вышегородцев Кирилл Евгеньевич,  
Оболенский Иван Александрович,  
Головня Максим Сергеевич (RU)**

(33) **RU**

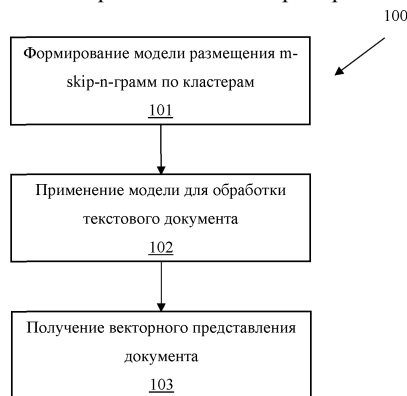
(71) Заявитель:

(74) Представитель:

**ПУБЛИЧНОЕ АКЦИОНЕРНОЕ  
ОБЩЕСТВО "СБЕРБАНК  
РОССИИ" (ПАО СБЕРБАНК) (RU)**

**Герасин Б.В. (RU)**

(57) Изобретение относится к вычислительным системам в широком смысле. Более конкретно - к системам и способам обработки естественного языка, искусственного языка, любых знаковых систем. Может использоваться в системах обработки информации, базах данных, электронных хранилищах. Техническим результатом является повышение скорости обработки документов на ЭВМ пользователя для классификации содержащейся в нем конфиденциальной информации, за счёт применения векторных представлений m-skip-n-грамм слов и их использовании для подсчёта частот предобученных семантических кластеров в текстовом документе для преобразования его в векторный вид. Заявленный технический результат достигается за счет выполнения компьютерно-реализуемого способа векторного представления электронного текстового документа для определения категории конфиденциальной информации, содержащейся в нем, выполняемого с помощью процессора и содержащего этапы, на которых: формируют по меньшей мере одну модель размещения m-skip-n-грамм по кластерам, при этом m-skip-n-грамма представляет по меньшей мере отдельное слово и при формировании упомянутой модели осуществляют определение списка используемых m-skip-n-грамм; преобразование в векторное представление каждой m-skip-n-граммы из списка; кластеризацию m-skip-n-грамм по их векторным представлениям; выполняют обработку по меньшей мере одного текстового документа с помощью полученной модели размещения m-skip-n-грамм, в ходе которой осуществляют подсчет встречаемости m-skip-n-грамм в текстовом документе; определяют кластеры текстового документа на основании встречаемости m-skip-n-грамм; суммируют количество встречаемости m-skip-n-грамм из каждого кластера; формируют векторное представление текстового документа на основании упорядоченной последовательности сумм m-skip-n-грамм; определяют категорию конфиденциальной информации в текстовом документе на основании модели размещения m-skip-n-грамм.



**A1**

**202192993**

**202192993**

**A1**

## **СПОСОБ И СИСТЕМА ПОЛУЧЕНИЯ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ ЭЛЕКТРОННОГО ТЕКСТОВОГО ДОКУМЕНТА ДЛЯ КЛАССИФИКАЦИИ ПО КАТЕГОРИЯМ КОНФИДЕНЦИАЛЬНОЙ ИНФОРМАЦИИ**

### **ОБЛАСТЬ ТЕХНИКИ**

[0001] Настоящее изобретение относится к вычислительным системам в широком смысле. Более конкретно – к системам и способам обработки естественного языка, искусственного языка, любых знаковых систем. Может использоваться в системах обработки информации, базах данных, электронных хранилищах.

### **УРОВЕНЬ ТЕХНИКИ**

[0002] Автоматическая обработка, передача, хранение документов, может включать классификацию исходных документов, их кластеризацию и иные действия путем соотнесения векторного представления документа с другим векторным представлением документа, любого множества или группы документов. Варианты реализации данного изобретения могут быть схожи с решениями, изложенными ранее в патентах RU 2701995 C2, RU 2583716 C2, RU 2254610 C2. Методы получения векторного представления и дальнейшей классификации – сети BERT, GPT, рекуррентные сети (RNN) и подобные, требуют больших вычислительных ресурсов. Вместе с этим подавляющее большинство пользователей имеют слабые персональные ЭВМ (стационарный ПК, ноутбуки), которые не содержат в себе специальных вычислительных устройств – многоядерных процессоров, графических вычислителей (GPU). Способы, которые не требуют больших мощностей (Tf-Idf), не обладают высокой эффективностью векторного представления документов для их дальнейшей обработки, например, классификации. Таким образом, в настоящее время, невозможно эффективно реализовывать получение векторного представления и/или классификацию документа на стороне пользователя.

[0003] Основные недостатки существующих решений обусловлены следующим:

- решения с высоким качеством классификации требуют специальных вычислительных устройств (многоядерных процессоров, GPU), которые отсутствуют у простых ПК;

- требуется хранить в памяти (оперативной или графической) всю предобученную модель нейронной сети. Стандартная версия «BERT» занимает 680 мегабайт (мультиязычная от Google: <https://huggingface.co/bert-base-multilingual-cased>; русскоязычная от DeepPavlov: <https://huggingface.co/DeepPavlov/rubert-base-cased-sentence>), расширенные версии 1,8 Гб (LaBSE: <https://huggingface.co/sentence-transformers/LaBSE>; от Сбербанка: [http://sberbank-ai/sbert\\_large\\_nlu\\_ru](http://sberbank-ai/sbert_large_nlu_ru));
- очень медленная скорость обработки для пользователя – несколько минут на текст из порядка  $10^3$  слов;
- ускорение обработки путём использования только части документа неминуемо ведёт к потере семантической целостности документа и неполноте информации о нём;
- организация работы на стороннем сервере требует передачи документа (часто и большого размера) и работы мощного вычислительного центра, что делает такой подход очень дороги, поскольку количество документов, подлежащих обработке, составляет десятки миллионов документов в месяц.

## **СУЩНОСТЬ ИЗОБРЕТЕНИЯ**

[0004] Заявленное изобретение направлено на решение технической проблемы, заключающейся в сокращении времени обработки текстовых данных на ЭВМ пользователя (офисный ПК пользователя) и возможности применения на неспециализированных ЭВМ с помощью их преобразовании в векторную форму без непосредственного использования моделей нейронных сетей.

[0005] Техническим результатом является повышение скорости обработки документов на ЭВМ пользователя для классификации содержащейся в нем конфиденциальной информации, за счёт применения векторных представлений  $m$ -skip- $n$ -грамм слов и их использовании для подсчёта частот предобученных семантических кластеров в текстовом документе для преобразования его в векторный вид.

[0006] Дополнительным результатом заявленного решения является также не только классификации документов на заданные категории конфиденциальной информации, но и возможность кластеризации документов внутри каждой категории, за счёт получаемого векторного вида документа, который имеет смысл частот семантических (смысловых) характеристик.

[0007] Заявленный технический результат достигается за счет выполнения компьютерно-реализуемого способа векторного представления электронного текстового

документа для определения категории конфиденциальной информации, содержащейся в нем, выполняемого с помощью процессора и содержащего этапы, на которых:

- формируют по меньшей мере одну модель размещения  $m$ -skip- $n$ -грамм по кластерам, при этом  $m$ -skip- $n$ -грамма представляет по меньшей мере отдельное слово и при формировании упомянутой модели осуществляют:
  - определение списка используемых  $m$ -skip- $n$ -грамм;
  - преобразование в векторное представление каждой  $m$ -skip- $n$ -граммы из списка;
  - кластеризацию  $m$ -skip- $n$ -грамм по их векторным представлениям;
- выполняют обработку по меньшей мере одного текстового документа с помощью полученной модели размещения  $m$ -skip- $n$ -грамм, в ходе которой:
  - осуществляют подсчет встречаемости  $m$ -skip- $n$ -грамм в текстовом документе;
  - определяют кластеры текстового документа на основании встречаемости  $m$ -skip- $n$ -грамм;
  - суммируют количество встречаемости  $m$ -skip- $n$ -грамм из каждого кластера;
  - формируют векторное представление текстового документа на основании упорядоченной последовательности сумм  $m$ -skip- $n$ -грамм;

определяют категорию конфиденциальной информации в текстовом документе на основании модели размещения  $m$ -skip- $n$ -грамм.

[0008] В одном частном варианте реализации способа используют нечеткое разбиение списка  $m$ -skip- $n$ -грамм на кластеры.

[0009] В другом частном варианте реализации способа каждая  $m$ -skip- $n$ -грамма относится к нескольким кластерам.

[0010] В другом частном варианте реализации способа каждая  $m$ -skip- $n$ -грамма имеет вес, характеризующий ее близость к заданному кластеру.

[0011] В другом частном варианте реализации способа кластеризация  $m$ -skip- $n$ -грамм по их векторным представлениям выполняется более одного раза.

[0012] В другом частном варианте реализации способа дополнительно используются веса для кластеров  $m$ -skip- $n$ -грамм, характеризующие значимость кластеров для векторного представления документа.

[0013] В другом частном варианте реализации способа дополнительно производится получение векторных представлений  $m$ -skip- $n$ -грамм, не входящих в список, и их соотнесение по векторным представлениям к по меньшей мере одному кластеру.

[0014] В другом частном варианте реализации способа список или часть списка используемых  $m$ -skip- $n$ -грамм формируется исходя из встречаемости  $m$ -skip- $n$ -грамм в текстовых данных, получаемых из внешних источников данных.

[0015] Заявленное решение также осуществляется с помощью системы получения векторного представления электронного документа, которая содержит по меньшей мере один процессор и по меньшей мере одну память, хранящую машиночитаемые инструкции, которые при их исполнении процессором реализуют вышеуказанный способ.

[0016] Заявленный технический результат также достигается за счет компьютерно-реализуемого способа определения категории конфиденциальной информации в текстовом документе, выполняемый с помощью процессора и содержащий этапы, на которых выполняют:

- предварительное формирование по меньшей мере одной модели размещения  $m$ -skip- $n$ -грамм по кластерам по заданной тематике, при этом  $m$ -skip- $n$ -грамма представляет по меньшей мере отдельное слово и при формировании упомянутой модели осуществляют:
  - определение списка используемых  $m$ -skip- $n$ -грамм;
  - преобразование в векторное представление каждой  $m$ -skip- $n$ -граммы из списка;
  - кластеризацию  $m$ -skip- $n$ -грамм по их векторным представлениям;
- получают по меньшей мере один электронный текстовый документ;
- выполняют обработку полученного текстового документа с помощью полученной модели размещения  $m$ -skip- $n$ -грамм, в ходе которой:
  - осуществляют подсчет встречаемости  $m$ -skip- $n$ -грамм в текстовом документе;
  - определяют кластеры текстового документа на основании встречаемости  $m$ -skip- $n$ -грамм;
  - суммируют количество встречаемости  $m$ -skip- $n$ -грамм из каждого кластера;
  - формируют векторное представление текстового документа на основании упорядоченной последовательности сумм  $m$ -skip- $n$ -грамм;
- определяют категорию конфиденциальной информации в электронном текстовом документе с помощью модели размещения  $m$ -skip- $n$ -грамм.

[0017] В настоящем документе под « $m$ -skip- $n$ -граммой слов» (или просто « $m$ -skip- $n$ -граммой» // [http://www.machinelearning.ru/wiki/images/7/78/2017\\_417\\_DrapakSN.pdf](http://www.machinelearning.ru/wiki/images/7/78/2017_417_DrapakSN.pdf)) понимается совокупность последовательности из  $n$  слов, которая получена из последовательности слов из некоторого текста, сохраняя в ней последовательность слов в текстах, при этом из исходной последовательности слов удалено  $m$  слов после каждого одного из  $n$  слов. Например, 0-skip-1-грамма слов это просто отдельные слова из текста. 0-skip-2-грамма это биграммы слов (пара подряд идущих слов в тексте), 0-skip-3-грамма это триграммы слов (тройка подряд идущих слов в тексте). Для построения 1-skip-2-граммы берется последовательность из трех слов, в которой удаляется второе слово - то есть, это первое и третье слово. Для построения 2-skip-4-граммы берется последовательность из 10 слов, в ней берется первое слово, затем 2 слова удаляется, затем берется следующее слово, далее удаляется 2 следующих слова, и так, пока не будет получена последовательность из 4 слов. Например, имеем предложение «в соответствии с одним или более вариантами реализации настоящего изобретения». В ней удаляются слова «соответствии с», «или более», «реализации настоящего». Тогда 2-skip-4-грамма для данного предложения будет выглядеть как «в одним вариантами изобретения». При этом при построении  $m$ -skip- $n$ -грамм за «слово» могут приниматься, как только слова языка, так и любой знак препинания, предлог, союз или любая самостоятельная единица языка. Всюду далее, без потери общности для упрощения изложения, под «словом» («словами» и прочим) будет понимать любую возможную  $m$ -skip- $n$ -грамму слов и последовательность  $m$ -skip- $n$ -грамм слов, если не сказано иное, например, «отдельных слов», «одиночных слов» и подобное. При этом также будем использовать и сам исходный термин  $m$ -skip- $n$ -грамма.

[0018] Под «эмбедингом» слова (от англ. embedding – вложение) или же «векторным представлением» слова или просто «вектором» слова будем понимать такой числовой вектор, который получен из слов или других языковых сущностей, и который определен для слова, и имеет фиксированную размерность для метода его получения. Другими словами, векторным представлением слова является упорядоченная последовательность чисел – числовой вектор некоторого размера, когда каждое слово имеет свой определенный числовой вектор. В самом простом случае эмбединги слов можно получить нумерацией слов в некотором словаре и постановкой значения равного 1 в векторе, размерность которого равна числу слов в этом словаре. При этом на остальных позициях будут находиться значения равные 0. Например, для русского языка можно использовать Толковый словарь Даля. В нём пронумеруем все слова от первого до последнего. Так слово «абажур» будет иметь значение 1 на позиции 3, «абанат» - иметь значение 1 на позиции 7, и так далее. Если в словаре 200000 слов, то эмбединг будет иметь размерность 200000.

Подобный метод построения эмбедингов называют – one-hot encoding. Описание изобретения не ограничивает способ получения векторов слов. Данные вектора могут быть получены, например, нейронной сетью, реализующей математическое преобразование из пространства с одним измерением на слово в некоторое пространство вектора с большей размерностью. Иными методами, позволяющими сопоставить каждой m-skip-n-грамме свой вектор чисел заданной размерности. Данное векторное представление слов можно получать из уже известных набор векторизированных слов (Word2Vec, Glove, FastText и другие), модифицируя их или без такового. При этом под кластеризацией понимается группировка множества объектов на подмножества (кластеры) таким образом, чтобы объекты из одного кластера были более похожи друг на друга, чем на объекты из других кластеров по заданному критерию. Под «весом» элемента, например, «вес кластера» или «вес m-skip-n-граммы», можно понимать математическую конструкцию, коэффициенты, множители, используемые при проведении суммирования, интегрирования или усреднения и прочего с целью придания некоторым элементам большей значимости в результирующем значении по сравнению с другими элементами. «Вес» можно определить и как дополнительный множитель, коэффициент или число, сопоставляемое отдельным слагаемым или другим факторам, в скалярном произведении элементов используемого векторного пространства.

## **КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ**

[0019] Настоящее изобретение иллюстрируется на примерах, без каких бы то ни было ограничений; его сущность становится понятной при рассмотрении приведенного ниже подробного описания изобретения в сочетании с чертежами, при этом:

[0020] На Фиг. 1 схематически показан пример процесса автоматического получения векторного представления документа.

[0021] На Фиг. 2 схематически показан пример процесса получения множества наборов кластеров с весами для каждого кластера в каждом наборе кластеров.

[0022] На Фиг. 3 схематически показан пример процесса получения множества m-skip-n-грамм слов, где каждая m-skip-n-грамм слова нечетко относится к каждому кластеру в каждом наборе кластеров.

[0023] На Фиг. 4 схематически показан пример процесса извлечения признаков документа.

[0024] На Фиг. 5 схематически показан пример сопоставления кластеров слов с позициями вектора документа, для двумерного случая векторного представления слов и четким соотношением слова к одному кластеру.

[0025] Фиг. 6 иллюстрирует общую схему вычислительного устройства.

## ОСУЩЕСТВЛЕНИЕ ИЗОБРЕТЕНИЯ

[0026] В настоящем документе описываются способы получения векторного представления электронного документа для дальнейшей обработки, передачи и хранения. Изобретение может быть применено к любым естественным языкам, искусственным языкам, любым знаковым системам, при этом, далее в настоящем документе, все это будем называть просто «языком». Таким образом, далее, под «языком» понимаются естественные языки, искусственные языки, любые знаковые системы.

[0027] Автоматическая обработка, передача, хранение документов, может включать классификацию исходных документов, их кластеризацию и иные действия путем сопоставления векторного представления документа с другим векторным представлением документа, любого множества или группы документов.

[0028] На Фиг. 1 представлена общая блок-схема заявленного способа получения векторной формы документа. На первом этапе (101) реализации изобретения производится получение модели размещения  $m$ -skip- $n$ -грамм по кластерам. Такая модель представляет собой список  $m$ -skip- $n$ -грамм каждая из которых соотносена по меньшей мере к одному из кластеров. Пример такой модели для частного случая, когда  $m$ -skip- $n$ -граммы являются символами и словами из русского языка, и разбивки на 1000 кластеров приведён в таблице ниже:

Таблица 1. Пример модели размещения  $m$ -skip- $n$ -грамм по кластерам

$m$ -skip- $n$ -грамма	кластер
!	737
,	737
.	737
этой	737
миллиметр	414
на	737
к	737
министр	457
спин	414
день	368
µm	414



президент	457
выступит	537
неделе	368

Процесс получения модели размещения m-skip-n-грамм по кластерам.

[0029] Для формирования модели сначала определяется список используемых m-skip-n-грамм. В частном случае это может быть толковый словарь языка, словари эмбедингов слов (Word2Vec, Glove, FastText и другие), список всех слов из статей с сайта «Википедия». В таком случае список слов должен быть большим, чтобы покрыть достаточное количество слов из анализируемых документов. В другом частном случае данный список проходит еще через нормализацию слов - процесс, когда получают леммы (<https://ru.wikipedia.org/wiki/Лемматизация>), стемминги (<https://ru.wikipedia.org/wiki/Стемминг>), морфологические корни, морфологические основы слов. В частном случае заглавные (прописные) и строчные буквы не различаются при формировании m-skip-n-грамм. В другом частном случае заглавные буквы различаются от строчных и их наличие образует разные m-skip-n-граммы. Помимо слов естественного, формального, искусственного языков в список могут включаться знаки препинания, цифры, аббревиатуры, нарицательные имена, имена собственные, сокращения, специальные символы (@” ;#№\$%”%^\*()\_}{\ и прочие), математические символы и любые другие печатные последовательности символов. В частном случае можно рассматривать любую последовательность печатных знаков в знаковой системе при введённом знаке разделения слов. Размер списка, конкретные m-skip-n-граммы, использование нормализации выбираются исходя из решаемой задачи, доступных вычислительных ресурсов и других факторов.

[0030] Далее каждая m-skip-n-грамма из списка преобразовывается в векторное представление с помощью модели естественного языка или без таковой. Настоящее описание не ограничивает способ получения, размерность или вид векторного представления. Для этого можно использовать способы, которые основаны на алгоритмах и моделях естественного языка Word2Vec, Glove, FastText, Universal Sentence Encoder, дополнение с трансформацией через сингулярное разложение, Tf-Idf. В одном частном варианте реализации вектор m-skip-n-граммы представляет собой конкатенацию нескольких векторов этой m-skip-n-граммы, полученных по разным способам.

[0031] В одном частном варианте реализации способа в качестве предобученной модели естественного языка используют по крайней мере одну из следующих нейронных

сетей: ELMO, GPT, GPT-2, GPT-3, LM-FiT, BERT-BASE, BERT-LARGE, RuBERT, CamemBERT, SciBERT, ROBERTA, R-Net, Semi-supervised Sequence learning, GloVe, Word2Vec, FastText, InferSent, OpenAI Transformer, USE-Transformer (Universal sentence encoder), USE-DAN, T-NLG (Turing Natural Language Generation) или наследуемые от них модели, в том числе и применение сокращений нейронных сетей (<https://blog.rasa.com/compressing-bert-for-faster-prediction-2/#overview-of-model-compression>) методами: дистилляции (<https://habr.com/ru/company/avito/blog/485290/>), прунинга (<https://neurohive.io/ru/novosti/kak-sokratit-razmer-nejroseti-na-10-20-i-ne-proigratv-tochnosti/>, <https://openreview.net/pdf?id=rJl-b3RcF7>), квантизации ([https://www.researchgate.net/publication/342938169\\_KVANTIZACIA\\_VESOV\\_GLUBOKIH\\_NEJRONNYH\\_SETEJ\\_POSLE\\_OBUCENIA](https://www.researchgate.net/publication/342938169_KVANTIZACIA_VESOV_GLUBOKIH_NEJRONNYH_SETEJ_POSLE_OBUCENIA), [https://www.hse.ru/data/2019/07/12/1477568723/Грачев\\_резюме.pdf](https://www.hse.ru/data/2019/07/12/1477568723/Грачев_резюме.pdf)).

[0032] В другом частном варианте реализации способа в качестве предобученной модели естественного языка используют вышеуказанные модели с последующем дообучением эмбедингов  $m$ -skip- $n$ -грамм на наборе текстов по банковским документам (данным, материалам) или ином наборе текстовых документов.

[0033] Далее выполняется кластеризация векторов  $m$ -skip- $n$ -грамм. Настоящее описание не ограничивает способ кластеризации списка  $m$ -skip- $n$ -граммы. Можно использовать такие алгоритмы кластеризации как K-Means, Affinity propagation, Mean-shift, Spectral clustering, Ward hierarchical clustering, Agglomerative clustering, DBSCAN, OPTICS, Gaussian mixtures, Birch. Конечным результатом кластеризации является соотнесение каждой  $m$ -skip- $n$ -граммы по меньшей мере к одному кластеру.

[0034] По завершению получения модели размещения  $m$ -skip- $n$ -грамм выполняется обработка текстовых документов на этапе (102) с ее применением. По завершению применения модели получаем векторный вид документа (103). Если остались еще необработанные документы, то переходим ко следующему документу. Так пока не будет обработан каждый документ.

[0035] На Фиг. 2 приведен пример этапов формирования модели размещения  $m$ -skip- $n$ -грамм по кластерам (101). Здесь осуществляется получение множества наборов кластеров с весами для каждого кластера в каждом наборе. В начале производим выбор базового списка  $m$ -skip- $n$ -грамм, затем получаем их вектора (1011). Настоящее описание не ограничивает как-либо список, вид  $m$ -skip- $n$ -грамм и значения параметров  $m$  и  $n$  в них. Отличительной частью изобретения является сама возможность использования не только отдельных, одиночных слов языка, но любых  $m$ -skip- $n$ -грамм слов. При этом не ограничивается возможность выбора таких  $n$  и  $m$ , что полученные  $m$ -skip- $n$ -граммы будут

совпадать, например, с отдельными словами, биграммami слов, триграммами слов и прочим.

[0036] Этот этап может реализовываться путём использования готовых словарей (Word2Vec, Glove, FastText и другие), определением всевозможных или только интересующих  $m$ -skip- $n$ -грамм исключительно в классе анализируемых текстовых документов (по имеющейся базе данных или учебном наборе) или любым иным способом. Далее получаем векторное представление выбранных  $m$ -skip- $n$ -грамм любым способом (алгоритмы и модели Word2Vec, Glove, FastText, Universal Sentence Encoder, дополнение с трансформацией сингулярным разложением, Tf-Idf и другие) (1011). На этапе 1012 проводим многократную кластеризацию векторного представления  $m$ -skip- $n$ -грамм слов. Данная кластеризация производится любым известным способом кластеризации объектов при котором изначально фиксируется требуемое количество кластеров. Отличительной особенностью изобретения является то, что кластеризация может проводиться несколько раз. При этом она может производиться на разное количество кластеров одним и тем же алгоритмом кластеризации, на одинаковое количество кластеров одним и тем же алгоритмом кластеризации, но с разными начальными инициализациями, на одинаковое количество кластеров различными алгоритмами кластеризации или любым иным способом получения различных совокупностей кластеризаций объектов.

[0037] Таким образом формируется множество наборов кластеров  $K = \{k_1, k_2, \dots, k_{N-1}, k_N\}$  (этап 1013). Для каждого  $C_{ij}$  кластера –  $j$ -ого кластера в  $i$ -ом разбиении на кластеры, где  $i = \overline{1, N}$ ,  $j = \overline{1, M_i}$ , задается свой весовой коэффициент  $q_{ij}$ , который может характеризовать значимость этого кластера для векторного представления документа (этап 1014). Описание изобретения не ограничивает методы получения весовых коэффициентов  $q_{ij}$  и их значения. Отличительной частью изобретения является сама возможность использования весовых коэффициентов для кластеров, что позволяет характеризовать значимость каждого кластера для векторного представления документа. Одной из возможных реализаций является подход, когда производится выделение «мусорных» кластеров и исключение данного кластера, например, обнулением его веса. При этом «мусорным» кластером могут являться кластеры с высокой долей «стоп-слов», общих слов, не несущих информативность слов для конкретной решаемой задачи анализа текстовых данных. Значения весовых коэффициентов можно получить какими-либо автоматизированными вычислениями, определить экспертной оценкой, любым иным способом. При этом возможна ситуация, когда все кластеры являются равнозначными или, как отмечено выше, когда какие-либо кластеры

вообще исключаются из использования. Результатом данного этапа является множество наборов кластеров с весами для каждого кластера в каждом наборе (этап 1015).

[0038] На Фиг. 3 приведен пример этапов получения нечеткого соотнесения слов из списка. Для всех или части слов из словаря проводим разбиение на кластеры  $N$  раз (этап 1013) следующим образом: для каждой кластеризации может выбираться определенное количество кластеров; выбирается некоторый метод кластеризации, который позволяет кластеризовать объекты на заданное или неопределенное количество кластеров, и производится кластеризация. Каждую кластеризацию обозначим  $k_i$ , где  $i = \overline{1, N}$ . Тогда общее множество наборов кластеров обозначим  $K = \{k_1, k_2, \dots, k_{N-1}, k_N\}$ . Каждая  $i$ -ая кластеризация  $k_i$  будет представлять собой набор из  $M_i$  кластеров. Каждый  $j$ -ый кластер в  $i$ -ом разбиении на кластеры обозначим  $C_{ij}$ , где  $i = \overline{1, N}$ ,  $j = \overline{1, M_i}$ . Далее, для данного множество наборов кластеров  $K = \{k_1, k_2, \dots, k_{N-1}, k_N\}$  выбираем показатели соотнесения слова к кластерам (этап 201). Показатель соотнесения слова к кластерам позволяет определить нечеткое распределение слов по кластерам. В качестве такого показателя может быть расстояние от  $m$ -skip- $n$ -грамм до центра кластера. Данное расстояние может нормироваться по расстояниям до центра всех кластеров, или по расстояниям только до нескольких (возможен вариант ближайших) кластеров. Другой возможный вариант реализации изобретения состоит в том, чтобы определять расстояние до оправленного количества ближайших объектов при четкой кластеризации и рассчитывать долю объектов из каждого кластера в этом оправленном количестве. При этом описание изобретения никак не ограничивает способ, меру и показатель получения нечеткого соотнесения объекта к кластеру. Отличительной особенностью изобретения является возможность использовать нечеткое соотнесение  $m$ -skip- $n$ -грамм к кластерам. Таким образом для каждого используемого  $s$ -го слова определяется некоторый показатель  $w^{(s)}_{ij}$ , который характеризует в какой степени  $s$ -ое слово относится к кластеру  $C_{ij}$  (этап 302).

[0039] Каждое слово мы можем отнести к любому количеству кластеров. Подобную разбивку на кластеры с нечетким разбиением по кластерам можно получить, например, методом нечеткой кластеризации «С-средних», или с использованием оценки специалиста. При этом могут быть выбраны такие показатели, что каждое слово будет относиться только к одному кластеру, например, в случае, когда значение показателя соотнесения равно единице для одного кластера (не обязательно самого близкого к вектору слова) и равно нулям для всех остальных. В некоторых случаях результаты такого выбора будут характеризовать четкую кластеризацию, но возможны варианты реализации изобретения, когда  $m$ -skip- $n$ -грамм не всегда будет относиться к самому близкому кластеру. То есть, при

этом результаты такого выбора показателей соотнесения, в общем случае, не обязательно будут соответствовать результатам работы алгоритмов четкой кластеризации слов. Описание изобретения не ограничивает количество разбинок на кластеры, которых, очевидно, должно быть больше нуля ( $>0$ ). Описание изобретения не ограничивает метрику и математические пространства для произведения кластеризации, и соотнесения слов с кластерами.

[0040] Этапы (201) и (202) имеют общие предшествующие шаги с этапами (1014) и (1015). Данными этапами являются этапы (1011, 1012, 1013). Описание изобретения не ограничивает взаимную выполнения упомянутых этапов шагов и их совокупное использование, а дает только пояснения к ним.

[0041] Пример раскрытия этапа (102) представлен на Фиг. 4. На вход этапа (102) подается документ с данными в текстовом виде. В частном варианте реализации способа сначала производится определение  $m$ -skip- $n$ -грамм, которые присутствуют в документе и которые присутствуют в списке  $m$ -skip- $n$ -грамм модели размещения (этап 1021).

[0042] При реализации этапа (1021) выполняется подсчет количества встречаемости  $m$ -skip- $n$ -грамм из сформированного списка в документе. Затем смотрятся к каким кластерам относятся эти  $m$ -skip- $n$ -граммы. Подсчитывается количество  $m$ -skip- $n$ -грамм из документа внутри каждого кластера.

Например, имеем следующее предложение: «министр выступит на этой неделе». В нём, в соответствии с примером из таблицы 1, кластер 457 встречается 1 раз («министр»), кластер 537 - 1 раз («выступит»), кластер 737 – 2 раза («на», «этой»), кластер 368 – 1 раз («неделе»). Тогда итоговый вектор документ будет представлять собой вектор, у которого на позициях 457, 537 и 368 будет стоять значение 1, на позиции 737 значение 2, на всех остальных позициях – значение 0. В другом частном варианте реализации способа используется последующая нормировку данного вектора. В рассмотренном примере  $m$ -skip- $n$ -граммы из списка встречаются 5 раз. Тогда нормированный вектор будет представлять собой вектор, у которого на позициях 457, 537 и 368 будет стоять значение 0,2, на позиции 737 значение 0,4, на всех остальных позициях – значение 0.

[0043] Далее начинается формирование вектора документа. Каждая позиция в этом векторе соответствует определенному кластеру. Сначала берётся  $m$ -skip- $n$ -грамма из документа, для последующего определения к какому кластеру она относится по сформированной модели размещения  $m$ -skip- $n$ -грамм. Затем значение, на соответствующей кластеру позиции в векторе документа, увеличиваем на количество этой  $m$ -skip- $n$ -граммы в документе (этап 1022). В одном из частных вариантов способа реализации значение в позиции вектора увеличиваем не на количество  $m$ -skip- $n$ -грамм, а на произведение

количества и веса  $m$ -skip- $n$ -граммы для этого кластера (этап 1022). На следующем этапе (1023) полученные значения на позициях вектора документа умножаем на соответствующие веса кластеров. Результатом является векторное представление документа (этап 1024).

[0044] В соответствии с одним или более вариантами реализации настоящего изобретения, пример способа автоматизированного получения векторного представления электронного документа может включать в себя следующие этапы. Выбирается вектор, размерность которого совпадает с общим количеством полученных кластеров во всех кластеризациях. Данный вектор инициализируем произвольными начальными значениями. Описание изобретения не ограничивает значения, используемые для начальной инициализации вектора.

[0045] Возможно также использовать, например, нулевые значения. Каждая позиция вектора строго соответствует определенному кластеру (Фиг. 5). Осуществляется извлечение слов из документа. Данное извлечение можно получить, например, последовательный проход по документу, использовать некоторое представление документа, которое уже имеет извлеченные слова и их количества, любым иным способом. Для каждого слова используют веса соотношения для каждого кластера в каждом наборе (этап 202).

[0046] Для примера рассмотрим последовательное извлечение слов из текста, не ограничивая варианты реализации изобретения. Выполняется получение слова из текста. По весам соотношения слова с каждым кластером выполняется поиск позиции в векторе документа, значения в которых необходимо увеличить. Осуществляем увеличение данных значений в выявленных позициях на некоторое значение. Это значение может быть фиксированным, либо изменяемым в зависимости от условий в процессе обработки документа. Данное значение уже может учитывать в себе вес соотношения слова с каждым кластером.

[0047] Возможен также вариант изобретения, когда увеличение значений на позиции в векторе происходит на фиксированное значение, которое затем умножается на вес соотношения слова. Описание настоящего решения не ограничивает методы изменения значений на позициях в векторе, которые связаны с данным словом. Возможно также учитывать в итоговом изменении значений в векторе нечеткое соотнесение слова к различным кластерам. При этом не ограничиваются значения изменений значений, которые, в общем случае, могут быть и отрицательными. Для расчета значений, на которые изменяются значения в позициях векторов, можно использовать различные методы, которые позволяют учитывать и любые иные характеристики слов. Таким примером может

быть использование метода «частоты использования слов - обратной частоты документа» (TF-IDF, Term Frequency - Inverse Document Frequency), или просто частоты слов. Также, для расчета значений, можно учитывать вес кластера, характеризующий значимость кластера для векторного представления документа. Проходя так по всему тексту, производим увеличение соответствующих позиций в векторе документа. Итогом прохода может стать векторный вид документа.

[0048] Настоящее описание не ограничивает подходы по использованию весов кластеров, характеризующие значимость каждого кластера для векторного представления документа, и весов соотношения слова для каждого кластера в каждом наборе. Заявленное решение дает возможность использовать данные веса при составлении векторного вида документа. Полученные значения могут использоваться для векторного вида документа или любое их множество и подмножество использоваться в обработке данных, в самостоятельном виде или в совокупности с другими данными, для получения нового векторного вида документа. Примером такого совокупного использования может служить конкатенация с вектором метода TF-IDF. В данном случае осуществляется составление векторного представления документа по методу TF-IDF. Составление вектора документа может осуществляться с помощью выполнения заявленного способа. Производится конкатенация двух векторов (операцию соединения, склеивания векторов). Результатом конкатенации и будет являться вектор документа. Дополнительно возможно дальнейшее проведение алгебраических преобразований над полученным вектором документа.

[0049] В одном частном варианте реализации способа используется нечеткое разбиение списка  $m$ -skip- $n$ -грамм на кластера. В этом варианте каждая  $m$ -skip- $n$ -грамм соотносится больше чем к одному кластеру. Каждая  $m$ -skip- $n$ -грамм имеет своё соотношение к кластеру в зависимости от расстояния до этого кластера. При этом для расчета данного расстояния могут использоваться и координаты центра кластера, и координаты  $m$ -skip- $n$ -грамм из этого и других кластеров. В частном случае такого способа кластеризации  $m$ -skip- $n$ -грамма относится ко всем кластерам. Примером алгоритма для реализации данным частном варианте кластеризации является C-Means.

[0050] В одном из частных вариантов реализации каждая  $m$ -skip- $n$ -грамма относится к нескольким кластерам. При этом  $m$ -skip- $n$ -грамма может относиться к одному или более кластерам в одинаковой степени.

[0051] В другом частном варианте реализации каждая  $m$ -skip- $n$ -грамма имеет вес, характеризующий ее соотношение к заданному кластеру. В отличие от предыдущих частных вариантов в данном случае вес может характеризовать не только близость  $m$ -skip- $n$ -граммы к кластеру. При расчете данного веса может учитываться, например, плотность кластеров,

чтобы  $m$ -skip- $n$ -грамму соотносить в большей степени к менее плотному близкому кластеру. Вес является множителем, который используется для каждой  $m$ -skip- $n$ -граммы с каждым кластером. Вес может принимать и нулевое значение.

[0052] Еще в одном частном варианте реализации кластеризация списка  $m$ -skip- $n$ -грамм по их векторным представлениям выполняется более одного раза. В таком варианте кластеризация производится несколько раз на разное количество кластеров одним и тем же алгоритмом кластеризации. Итоговый вектор документа будет представлять собой конкатенацию векторов по каждой из кластеризации (пример на Фиг. 5). В другом случае используются разные алгоритмы кластеризации для разбиения на одинаковое количество кластеров или на разное количество кластеров. В еще одном случае используется одинаковое количество кластеров, и один и тот же алгоритм, но с разными начальными инициализациями, если алгоритм подразумевает возможность различного результата при разных инициализациях (K-Means, C-Means, Spectral clustering, Gaussian mixtures и другие). Количество кластеризаций, количество кластеров в каждой кластеризации и алгоритм кластеризации подбирается в зависимости от решаемой задачи. В частном случае, если список  $m$ -skip- $n$ -грамм более 10000, можно использовать разбивку на кластера размерами: 50, 100, 200, 300, 500, 700, 1000, 1500, 2000, 3000, 5000. В ином частном случае составляются отдельные списки, например, для 0-skip-1-грамм (одиночных слов), 0-skip-2-грамм (биграмм слов), 0-skip-3-грамм (триграмм слов). Каждый из этих списков отдельно кластеризуется несколько раз. Итоговым вектором документа будет являться конкатенация векторов по каждой из кластеризации для каждого списка.

[0053] В другом частном варианте реализации для каждого кластера  $m$ -skip- $n$ -грамм в каждой кластеризации (если их несколько) используются веса. Эти веса характеризуют значимость кластеров для векторного представления документа. Веса могут иметь и нулевые значения. В таком случае количество встречаемости  $m$ -skip- $n$ -грамм в этом кластере обнуляется. Этот подход полезен, когда требуется исключить из вектора кластера, которые содержат стоп-слова – слова, не несущие тематического смысла (и, к, у, о, при, на и прочие). В частных случаях веса для кластеров могут рассчитываться методами машинного обучения, задаваться как экспертная оценка.

[0054] Также,  $m$ -skip- $n$ -грамма может расширяться новыми  $m$ -skip- $n$ -граммами. В таком варианте выбирается способ расчета вектора  $m$ -skip- $n$ -граммы. Если в документ встречается  $m$ -skip- $n$ -грамма, которой нет в списке, то для нее рассчитывается векторное представление (вектор). Далее этот вектор соотносится к кластерам, которые получены на этапе кластеризации списка  $m$ -skip- $n$ -грамм. При этом новая  $m$ -skip- $n$ -грамма соотносится



по меньшей мере к одному кластеру. Дальнейшие шаги способа аналогичны случаю, если эта  $m$ -skip- $n$ -грамма присутствует в модели размещения  $m$ -skip- $n$ -грамм по кластерам.

[0055] Список используемых  $m$ -skip- $n$ -грамм может формироваться исходя из встречаемости  $m$ -skip- $n$ -грамм в текстовых данных, получаемых из внешних источников данных. Таким множеством текстов может быть любой внешний массив текстовых данных. Это может быть, например, множество анализируемых текстов различной тематики, собираемых, например, через новостные сайты. Обработка, классификация и кластеризация собираемой информации позволяет найти семантические сходства, аналоги, реализовать ранжирование результатов поиска или решить любую другую задачу обработки языка. Применение заявленного решения не ограничивает источник и природу используемого множества тестовых данных.

[0056] Примером реализации такого варианта является случай решения задачи классификации текстовых документов. Тогда список  $m$ -skip- $n$ -грамм формируется по исходному набору документов. Например, берутся все одиночные слова и биграммы, которые встречаются в этих текстах. Или 200000 самых частых биграмм. Для этих списков рассчитываются их векторные представления. Далее эти вектора подаются на кластеризацию, которая реализуется описанными способами.

[0057] При реализации настоящего изобретения все описанные варианты можно использовать в любой возможной совокупности и сочетании. Примером такого частного варианта является случай, когда по имеющемуся набору данных выбираются все встречающиеся одиночные слова, биграмм слов, триграмм слов. В каждом списке рассчитываются вектора  $n$ -грамм слов. Далее каждый список векторов подаётся на многократную кластеризацию. В каждой кластеризации для каждой  $n$ -граммы выбирается вес соотнесения ее к кластеру. Затем берём каждый анализируемый документ. По встречаемости  $n$ -грамм получаем вектор по каждой кластеризации. Значения в кластерах умножаются на веса кластеров. Каждый такой вектор нормируется. Все вектора конкатенируются в общий вектор. Описание патента не ограничивает совокупности использования различных подходов.

[0058] Отличительной особенностью изобретения является предоставление возможности учёта ранее неизвестных  $m$ -skip- $n$ -грамм. Не ограничивая изобретение можно привести следующий подход для учёта новых  $m$ -skip- $n$ -грамм. Например, проходя по текстовому представлению документа при выявлении  $m$ -skip- $n$ -граммы, которой нет в используемом словаре, формируется последующее векторное представление данной  $m$ -skip- $n$ -граммы. После чего  $m$ -skip- $n$ -грамма относится к определенному кластеру, центр которого является ближайший к ней.

[0059] Ускорение работы вычислительного алгоритма достигается за счёт того, что при реализации заявленного решения непосредственно не используется нейронная сеть (GPT, BERT и прочие). Не требуется пропускать текст документа через огромное множество слоёв нейронной сети, что и обуславливает длительное время работы нейронной сети в несколько минут. Сама нейронная сеть применяется на этапе обучения для получения эмбедингов  $m$ -skip- $n$ -грамм. Эти эмбединги включают в себя высокое качество получения векторного представления фраз для данной нейронной сети. Далее производится их объединение в различные семантические группы путём многообразной кластеризации. Таким образом, используемая при функционировании настоящего решения обученная модель представляет собой простые словари  $m$ -skip- $n$ -грамм, где каждой  $m$ -skip- $n$ -грамм соответствует номер кластера. А при получении векторного представления документа осуществляется поиск  $m$ -skip- $n$ -грамм в словаре.

[0060] При использовании хранения словарей  $m$ -skip- $n$ -грамм, например, в виде хешированных таблиц, сложность такого поиска составляет  $O(1)$  и является самым быстрым поиском в общем случае. Поскольку при реализации заявленного решения используется не целиком текст документа, как в случае нейронных сетей, а его разбиение на  $m$ -skip- $n$ -граммы, то возможно некоторое падение точности анализа. Однако падение крайне мало и несущественно за счет того, что используются важные, семантически значимые  $m$ -skip- $n$ -граммы, которые объединяются в многообразный семантический кластер, частоты которых и представляют векторный вид документа, тем самым сохраняя большинство семантических связей слов в документе. Таким образом достигается сокращение время обработки текста, содержащий порядка  $10^4$  слов, с нескольких минут (порядка  $10^2$  секунд), до десятых долей секунд (порядка  $10^{-1} - 1$ ). Ускорение вычислений может доходит до 1000 раз. При этом падение точности классификации является незначительным и приемлемым при таком увеличении скорости обработки. Малое падение точности отличает изобретение от методов с использованием известных методов классификации на основе только Tf-Idf.

[0061] Также, заявленное решение позволяет снизить потребность в оперативной памяти. Для работы нейронной сети требует порядка  $10^2$ - $10^3$  Мбайт памяти (оперативной или видео памяти). Как указано ранее в описании нейронные сети могут занимать и 680 Мбайт (BERT) и 1,8 Гбайт (LaBSE). Модель изобретения представляет собой словари  $m$ -skip- $n$ -граммы с номерами кластеров. Для 100000 слов (0-skip-1-граммы) такой словарь занимает в памяти чуть больше 1 Мбайта. Для 100000 биграмм ( $m$ -skip-2-граммы) менее 3 Мбайт. Таким образом, даже при использовании по 10 различных

кластеризаций для слов, биграмм и триграмм общая модель будет занимать менее 100Мбайт, что в несколько раз меньше моделей нейронных сетей.

[0062] Заявленное решение может быть реализовано на общедоступных ЭВМ. Поскольку этапы получения векторного вида текстового документа представляют собой поиск в таблицах и простые операции нормировки вектора, то для данных операций не требуется специальных графических вычислителей (GPU). Функционирование изобретения не требуются огромного множества матричных вычислений, как в случае функционирования нейронных сетей, а способно эффективно выполняться на центральном процессоре (CPU) ЭВМ. Получение векторного вида и выполнение классификации электронного документа занимает порядка  $10^{-1}$  секунд даже на офисных персональных ЭВМ (ноутбуках) выпуском до 5 лет назад.

[0063] В заявленном решении также существует возможность разделения этапа получения векторного вида электронного документа и этапа его классификации. Векторный вид документа возможно получать на ЭВМ пользователя. Сформированный вектор можно отправлять на сервер для проведения централизованной классификации. При этом размер такого вектора составляет 1-10 Кбайт. На сервере проводится классификация данного документа по располагающейся там модели классификации (модели размещения m-skip-n-грамм). Это позволяет централизованно выполнять обновления модели классификации. Реализация такого подхода эффективно невозможна в случае функционирования нейронных сетей, поскольку на стороне пользователя производится расчёт по всей модели, которая имеет неотделимые (неразрывные) части получения векторного представления и классификации. Выделения из нейронной сети отдельных слоев классификации и их вывод на сервер не имеет смысла, поскольку они прочно связаны логикой работы и со значениями всей нейронной сети. Эффективное переобучение слоёв кластеризации возможно только при переобучении всей сети. Описываемый подход централизованного функционирования классификатора на сервере возможно реализовать если отсылать весь текст документа на сервер. Но размер таких данных может составлять и более 100 Мбайт и требует высоких вычислительных мощностей на сервере, в том числе и графических.

[0064] При реализации настоящего решения появляется возможность обработки всего документа и в случае размера его содержания порядка  $10^5$  слов. Методы, основанные на нейронных сетях, очень долго обрабатывают длинные документы. Документ с содержанием в 100000 слов могут обрабатываться в течении десятков минут. Для определённых нейронных сетей требуется равномерное распределение смыслового содержания документа, поскольку конечное содержание будет являться более значимым. Время

обработки нелинейно возрастает от размера документа. Это связано с архитектурами внутренних слоёв нейронных сетей, использовании им рекуррентных подходов, или использовании «механизма внимания» (англ. attention mechanism, attention model) в архитектуре-трансформере. Для возможности обработки таких документов в приемлемое время берётся только часть документа. Обычно такая часть составляет порядка 1000 слов. Недостатком такого подхода является то, что подчасть документа может быть непрезентабельной и не соответствовать смыслу и содержанию всего документа. Описываемое изобретение имеет линейную зависимость времени обработки от размера документа. Может обрабатывать весь документ, учитывая семантику каждой подчасти документа. Это достигается тем, что учитывается каждая используемая  $m$ -skip- $n$ -грамма, а время обработки зависит только от их количества. Таким образом, время обработки документа с содержанием порядка  $10^5$  будет составлять порядка 1 секунды. А в получаемом векторном представлении содержание начала документа будет равнозначно содержанию его середины или его концовки.

[0065] Заявленный способ (100) может применяться в частности для поиска и отбора схожих банковских документов или для кластеризации. Эффектом работы изобретения, после классификации документа (текстовой информации) как содержащий конфиденциальную информацию, может быть одно или несколько действий из следующих:

- блокировка ЭВМ и иных доступов пользователя, если пользователь не имеет права обработки или ознакомления с конфиденциальной информацией;
- сбор с ЭВМ пользователя, или другой электронной системы, данных о его учётных записях, ip-адресе, сетевом окружении, геолокации, используемом оборудовании, программном обеспечении или иных данных, позволяющих идентифицировать пользователя, с дальнейшей пересылкой собранных данных и сигнала о требуемом внимании в уполномоченное подразделение для проведения проверочных мероприятий и расследований;
- блокировка пересылки электронного сообщения и/или документов, если содержащийся в нём контент или вложение содержит конфиденциальную информацию, а отправка осуществляется на внешний почтовый ящик (вне защищаемого периметра организации), или в адресатах есть сотрудники, не имеющие доступа к конфиденциальной информации с учетом установленных политик доступа.

[0066] По итогам классификации конфиденциальной информации, управляющее воздействие в отношении ЭВМ пользователя может также включать непосредственную блокировку ЭВМ с оповещением об установленном нарушении, например, с помощью

отображения соответствующего уведомления на экране АРМ и формировании пакетов данных или сигнала на ЭВМ сотрудника службы безопасности. Каждый факт нарушения установленного доступа к конфиденциальной информации может сохраняться в базе данных для ведения журнала учета. Представленные примеры являются лишь частными случаями формирования итогового воздействия при реализации заявленного решения и не ограничивают иные возможные примеры его использования для целей предотвращения утечки чувствительных данных.

[0067] На Фиг. 6 представлен общий вид вычислительного устройства (400), пригодного для реализации заявленного решения. Устройство (400) может представлять собой, например, сервер или иной тип вычислительного устройства, который может применяться для реализации заявленного технического решения. В том числе входит в состав облачной вычислительной платформы/

[0068] В общем случае вычислительное устройство (400) содержит объединенные общей шиной информационного обмена один или несколько процессоров (401), средства памяти, такие как ОЗУ (402) и ПЗУ (403), интерфейсы ввода/вывода (404), устройства ввода/вывода (405), и устройство для сетевого взаимодействия (406).

[0069] Процессор (401) (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. В качестве процессора (501) может также применяться графический процессор, например, Nvidia, AMD, Graphcore и пр.

[0070] ОЗУ (402) представляет собой оперативную память и предназначено для хранения исполняемых процессором (401) машиночитаемых инструкций для выполнения необходимых операций по логической обработке данных. ОЗУ (402), как правило, содержит исполняемые инструкции операционной системы и соответствующих программных компонент (приложения, программные модули и т.п.).

[0071] ПЗУ (403) 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.

[0072] Для организации работы компонентов устройства (400) и организации работы внешних подключаемых устройств применяются различные виды интерфейсов В/В (404). 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.

[0073] Для обеспечения взаимодействия пользователя с вычислительным устройством (400) применяются различные средства (405) 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.

[0074] Средство сетевого взаимодействия (406) обеспечивает передачу данных устройством (400) посредством внутренней или внешней вычислительной сети, например, Интранет, Интернет, ЛВС и т.п. В качестве одного или более средств (406) 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.

[0075] Дополнительно могут применяться также средства спутниковой навигации в составе устройства (400), например, GPS, ГЛОНАСС, BeiDou, Galileo.

[0076] Таким образом, заявленное решение позволяет достичь следующих преимуществ:

- Повысить скорость обработки текстовых данных банковского документа.
- Уменьшить ресурсозатраты по оперативной памяти на реализацию обработки текстовых данных банковского документа.
- Реализовать обработку документа на ЭВМ пользователя с приемлемым временем работы и высоким качеством классификации.
- Возможность обработки очень больших документов (порядка  $10^5$  слов) за время порядка секунды.
- Реализовать обработку на общедоступных персональных ЭВМ только с центральным процессором (CPU), без использования графических процессоров (GPU).
- Повысить сохранность семантического смысла путём кластеризации m-skip-n-грамм слов, а не только отдельных слов как производится в известных патентах.
- Повысить точность представления текста с помощью латентных тематик через осуществление многократной кластеризация одних и тех же m-skip-n-грамм слов, в отличии от известных патентов, где нигде не указано, что может производиться

многократная или повторная кластеризация. В подходе нашего патента, например, можно производить кластеризация на 100, 200, 357, 500, 500 (снова на 500, но с новой инициализацией), 1000, 5000, 10000 кластеров.

- Сохранить различную семантику слов, производя сопоставление слов к нескольким кластерам, тем самым сохраняется неоднозначность терминов, в отличие от известных патентов, где имеем отнесение каждого слова только к одному (ближайшему) кластеру.
- Путём использования весов (априорных или апостериорных) для кластеров возможно снизить значимость или вообще исключить кластера с общими словами, тем самым повысить качество анализа, в частности расчёта близости двух текстов, в отличие от известных патентов, где нет никаких весов кластеров.
- Повысить точность анализа через обработку новых, ранее неизвестных слов.
- Расширить применение получаемого векторного представления и непосредственное использование получаемого вектора, без обучения классификаторов, для задач выявления семантических сходств, поиска аналогов, ранжирование результатов поиска и прочих.

[0077] 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

1. Компьютерно-реализуемый способ векторного представления электронного текстового документа для определения категории конфиденциальной информации, содержащейся в нем, выполняемый с помощью процессора и содержащий этапы, на которых:

- формируют по меньшей мере одну модель размещения  $m$ -skip- $n$ -грамм по кластерам, при этом  $m$ -skip- $n$ -грамма представляет по меньшей мере отдельное слово и при формировании упомянутой модели осуществляют:
  - определение списка используемых  $m$ -skip- $n$ -грамм;
  - преобразование в векторное представление каждой  $m$ -skip- $n$ -граммы из списка;
  - кластеризацию  $m$ -skip- $n$ -грамм по их векторным представлениям;
- выполняют обработку по меньшей мере одного текстового документа с помощью полученной модели размещения  $m$ -skip- $n$ -грамм, в ходе которой:
  - осуществляют подсчет встречаемости  $m$ -skip- $n$ -грамм в текстовом документе;
  - определяют кластеры текстового документа на основании встречаемости  $m$ -skip- $n$ -грамм;
  - суммируют количество встречаемости  $m$ -skip- $n$ -грамм из каждого кластера;
  - формируют векторное представление текстового документа на основании упорядоченной последовательности сумм  $m$ -skip- $n$ -грамм;

определяют категорию конфиденциальной информации в текстовом документе на основании модели размещения  $m$ -skip- $n$ -грамм.

2. Способ по п. 1, отличающийся тем, что используют нечеткое разбиение списка  $m$ -skip- $n$ -грамм на кластера.

3. Способ по п. 1, отличающийся тем, что каждая  $m$ -skip- $n$ -грамма относится к нескольким кластерам.

4. Способ по п. 1, отличающийся тем, что каждая  $m$ -skip- $n$ -грамма имеет вес, характеризующий ее соотношение к заданному кластеру.

5. Способ по п. 1, отличающийся тем, что кластеризация  $m$ -skip- $n$ -грамм по их векторным представлениям выполняется более одного раза.



6. Способ по п. 1, отличающийся тем, что дополнительно используются веса для кластеров  $m$ -skip- $n$ -грамм, характеризующие значимость кластеров для векторного представления документа.

7. Способ по п. 1, отличающийся тем, что дополнительно производится получение векторных представлений  $m$ -skip- $n$ -грамм, не входящих в список, и их соотнесение по векторным представлениям к по меньшей мере одному кластеру.

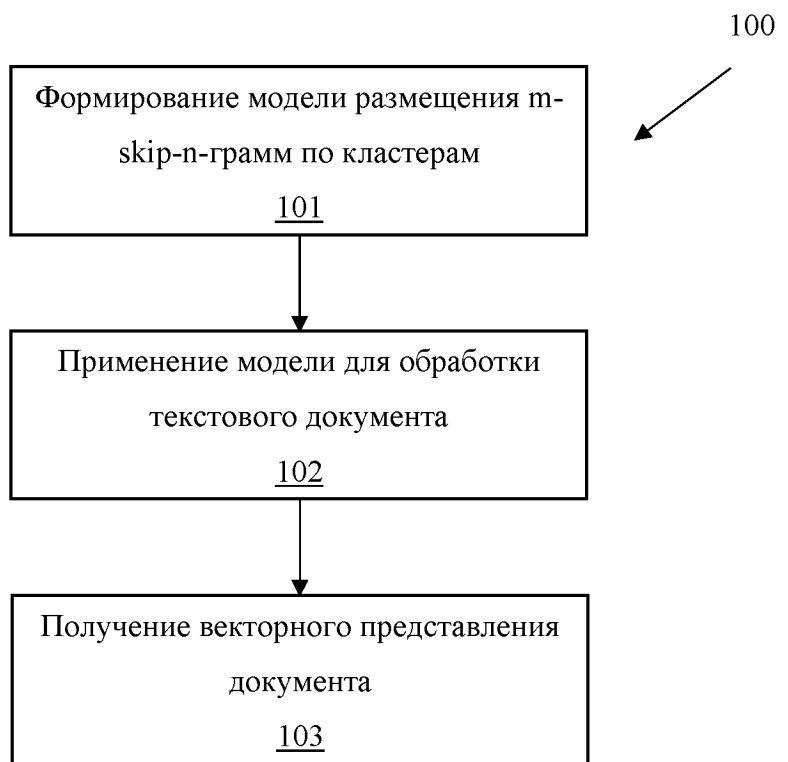
8. Способ по п. 1, отличающийся тем, что список или часть списка используемых  $m$ -skip- $n$ -грамм формируется исходя из встречаемости  $m$ -skip- $n$ -грамм в текстовых данных, получаемых из внешних источников данных.

9. Система получения векторного представления электронного документа, содержащая по меньшей мере один процессор и по меньшей мере одну память, хранящую машиночитаемые инструкции, которые при их исполнении процессором реализуют способ по любому из пп. 1-8.

10. Компьютерно-реализуемый способ определения категории конфиденциальной информации в текстовом документе, выполняемый с помощью процессора и содержащий этапы, на которых:

- выполняют предварительное формирование по меньшей мере одной модели размещения  $m$ -skip- $n$ -грамм по кластерам по заданной тематике, при этом  $m$ -skip- $n$ -грамма представляет по меньшей мере отдельное слово и при формировании упомянутой модели осуществляют:
  - определение списка используемых  $m$ -skip- $n$ -грамм;
  - преобразование в векторное представление каждой  $m$ -skip- $n$ -граммы из списка;
  - кластеризацию  $m$ -skip- $n$ -грамм по их векторным представлениям;
- получают по меньшей мере один электронный текстовый документ;
- выполняют обработку полученного текстового документа с помощью полученной модели размещения  $m$ -skip- $n$ -грамм, в ходе которой:
  - осуществляют подсчет встречаемости  $m$ -skip- $n$ -грамм в текстовом документе;
  - определяют кластеры текстового документа на основании встречаемости  $m$ -skip- $n$ -грамм;
  - суммируют количество встречаемости  $m$ -skip- $n$ -грамм из каждого кластера;
  - формируют векторное представление текстового документа на основании упорядоченной последовательности сумм  $m$ -skip- $n$ -грамм;

- определяют категорию конфиденциальной информации в электронном текстовом документе с помощью модели размещения  $m$ -skip- $n$ -грамм.



**Фиг. 1**



**Фиг. 2**

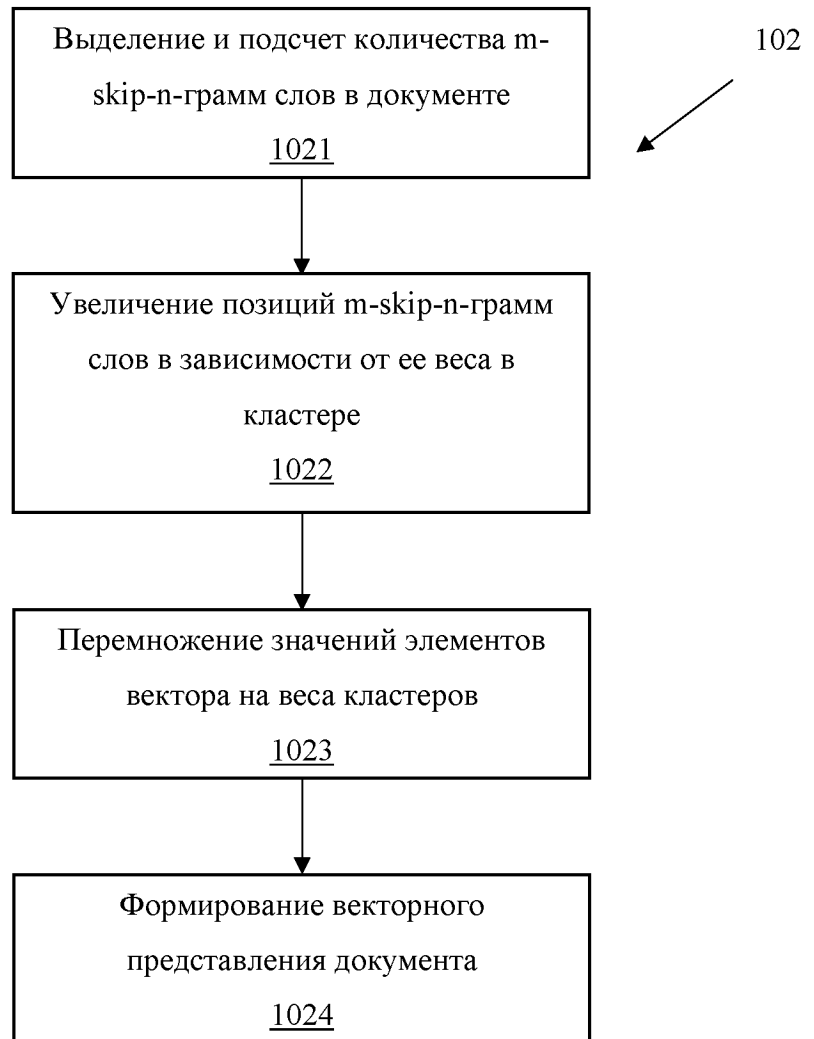
200



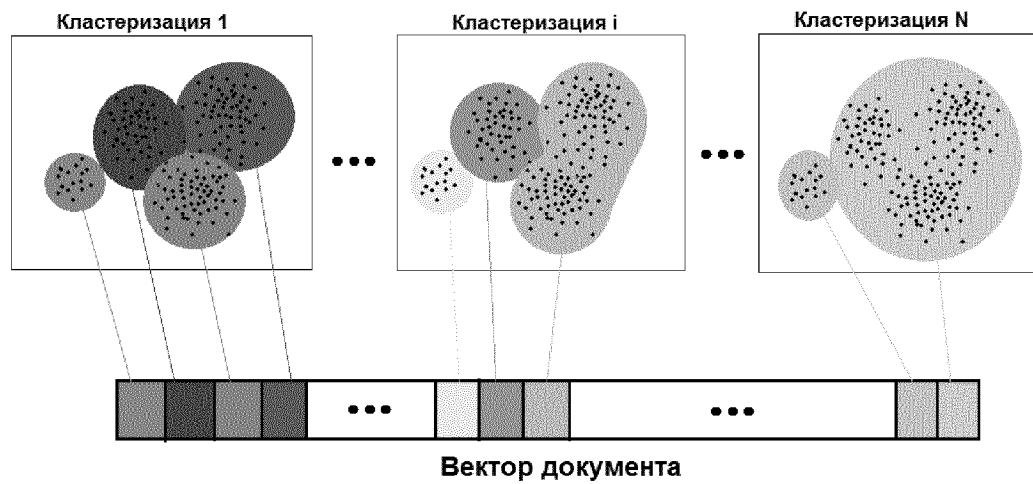
Задание показателя соотношения  
каждой  $m$ -skip- $n$ -грамм слов к  
каждому кластеру в каждом наборе  
201

Получение множества  $m$ -skip- $n$ -грамм  
слов с весами соотношения для  
каждого кластера в каждом наборе  
202

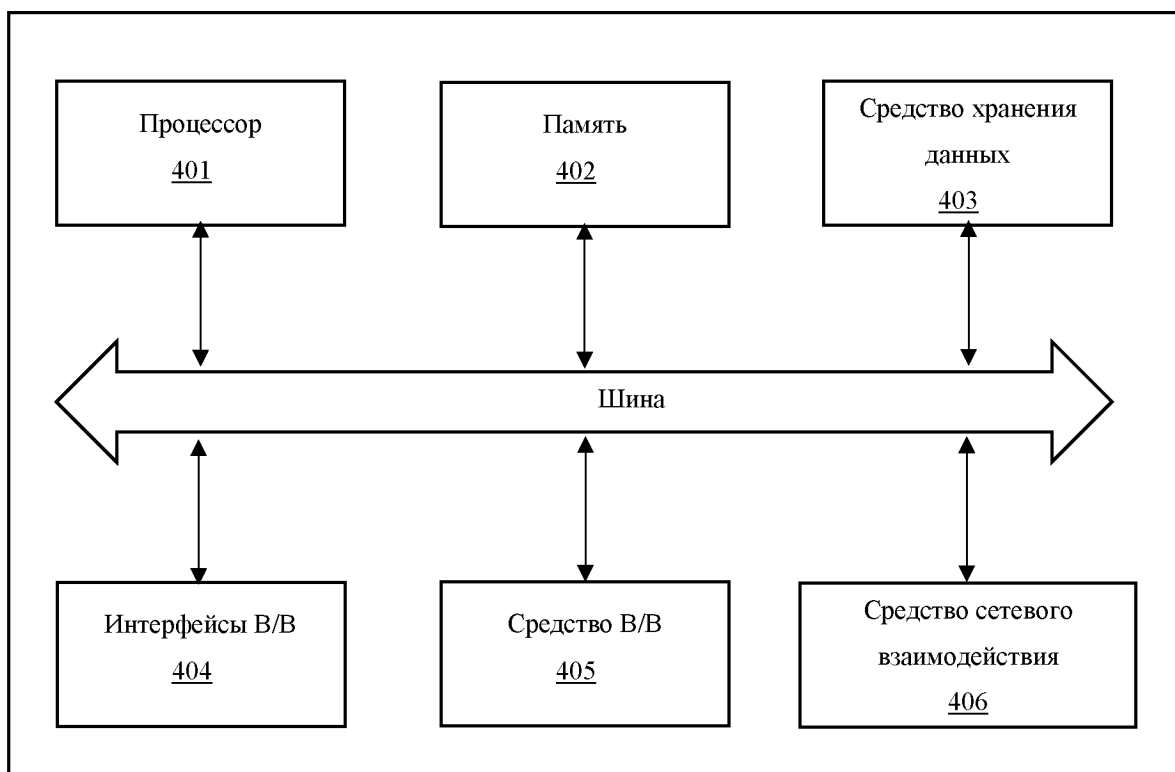
**Фиг. 3**



**Фиг. 4**



Фиг. 5



Фиг. 6

**ОТЧЕТ О ПАТЕНТНОМ ПОИСКЕ**

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

Номер евразийской заявки:

**202192993****А. КЛАССИФИКАЦИЯ ПРЕДМЕТА ИЗОБРЕТЕНИЯ:**G06F 16/35 (2019.01)  
G06F 40/284 (2020.01)  
G06F 40/30 (2020.01)

Согласно Международной патентной классификации (МПК)

**Б. ОБЛАСТЬ ПОИСКА:**

Просмотренная документация (система классификации и индексы МПК)

G06F 16/00, 16/30, 16/35, 16/50, 16/55, 17/00, 17/40, 40/00, 40/20, 40/279, 40/284, 40/30, G06K 9/00, G06N 7/00, 99/00, H04L 12/00

Электронная база данных, использовавшаяся при поиске (название базы и, если, возможно, используемые поисковые термины)  
Espacenet, ЕАПАТИС, ЕРОQUE Net, Reaxys, Google**В. ДОКУМЕНТЫ, СЧИТАЮЩИЕСЯ РЕЛЕВАНТНЫМИ**

Категория*	Ссылки на документы с указанием, где это возможно, релевантных частей	Относится к пункту №
A	US 9852337 B1 (OPEN TEXT CORPORATION) 26.12.2017	1-10
A	US 9280747 B1 (SAS INSTITUTE INC.) 08.03.2016	1-10
A	US 10891699 B2 (LEGALOGIC LTD.) 12.01.2021	1-10
A	RU 2634180 C1 (АКЦИОНЕРНОЕ ОБЩЕСТВО "ЛАБОРАТОРИЯ КАСПЕРСКОГО") 24.10.2017	1-10
A	RU 2701995 C2 (ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ "АБИ ПРОДАКШН") 02.10.2019	1-10
A	RU 2583716 C2 (ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ "АБИ ИНФОПОИСК") 10.05.2016	1-10

 последующие документы указаны в продолжении

\* Особые категории ссылочных документов:

«А» - документ, определяющий общий уровень техники

«D» - документ, приведенный в евразийской заявке

«Е» - более ранний документ, но опубликованный на дату подачи евразийской заявки или после нее

«О» - документ, относящийся к устному раскрытию, экспонированию и т.д.

"P" - документ, опубликованный до даты подачи евразийской заявки, но после даты испрашиваемого приоритета"

«Т» - более поздний документ, опубликованный после даты приоритета и приведенный для понимания изобретения

«Х» - документ, имеющий наиболее близкое отношение к предмету поиска, порочащий новизну или изобретательский уровень, взятый в отдельности

«У» - документ, имеющий наиболее близкое отношение к предмету поиска, порочащий изобретательский уровень в сочетании с другими документами той же категории

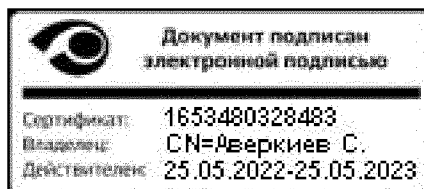
«&amp;» - документ, являющийся патентом-аналогом

«L» - документ, приведенный в других целях

Дата проведения патентного поиска: 07 июля 2022 (07.07.2022)

Уполномоченное лицо:

Начальник Управления экспертизы



С.Е. Аверкиев