УДК 004.82, 004.89

М. С. Куприянов, И. И. Холод, А. В. Шоров, Ю. А. Шичкина
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Информационные системы интеллектуального анализа данных и процессов (проблема Big Date)

Описывается подход к построению облачной платформы для интеллектуального анализа данных. Рассматривается подход к декомпозиции алгоритмов на функциональные блоки, позволяющий распределять их выполнение на отдельные узлы. Приводится описание архитектуры платформы и место в ней библиотеки алгоритмов интеллектуального анализа данных, реализующей рассмотренный подход.

Интеллектуальный анализ данных, облако, большие данные

Актуальность и проблема. С появлением первых ЭВМ наступил этап информатизации разных сторон человеческой деятельности. Если раньше человек основное внимание уделял веществу, затем энергии, то сегодня можно без преувеличения сказать, что наступил этап осознания процессов, связанных с информацией. Вычислительная техника создавалась прежде всего для обработки данных. В настоящее время современные вычислительные системы и компьютерные сети позволяют накапливать большие массивы данных для решения задач обработки и анализа. К сожалению, сама по себе машинная форма представления данных содержит информацию, необходимую человеку, в скрытом виде, и для ее извлечения нужно использовать специальные методы анализа данных [1]–[7].

Данная потребность стимулирует бурное развитие технологий интеллектуального анализа. Основным преимуществом этой технологии является извлечение из данных более сложных (по сравнению с математической статистикой) видов закономерностей, скрытых в данных, но более понятных человеку. Наибольший эффект от методов интеллектуального анализа достигается при обработке *больших данных*. Однако их анализ является ресурсоемкой задачей. Учитывая бурный рост технологий распределенных вычислений, и в частности облачных вычислительных сред, естественным является интеграция технологий: интеллектуального анализа данных, обработки больших данных, распределенных и облачных вычислений.

Компания Gartner в своем отчете от 24 сентября 2014 г. [8] отмечает, что инвестиции в технологии анализа больших данных будут расширяться. Так недавний обзор компании показал, что 73 % респондентов планируют увеличить инвестиции в большие данные в течение следующих двух лет. Кроме того, обзор показал, что количество организаций, которые не планируют работать с большими данными, уменьшился с 31 % в 2013 г. до 24 % в 2014 г.

Онлайн-издание Enterprise Management Quarterly (EMQ), принадлежащее компании Boston Hannah International, разместило статью [9], где описаны возможности, которые могут быть получены налоговыми органами при использовании методов интеллектуального анализа данных в системах облачных вычислений.

Онлайн-сообщество InformationWeek опубликовала статью [10], где отмечается, что, хотя сегодня накапливается большое количество данных из разных источников и бизнес часто видит в больших данных некоторый инструмент, позволяющий решить все проблемы, не нужно воспринимать это как панацею. Другими словами, современные инструменты не всегда позволяют достичь необходимого результата затрачивая разумные средства.

Журнал New York Time приводит 9 проблем больших данных [11]. В частности, отмечается, что анализ подобных данных очень сложен и современные алгоритмы анализа зачастую дают не тот результат, который необходим пользователю. Это приводит к разочарованию пользователей в механизмах анализа данных и к необходимости искать новые методы обработки больших данных.

Таким образом, несмотря на интенсивное развитие этих технологий, нерешенными проблемами сегодня остаются:

– адаптация существующих последовательных алгоритмов к распределенному выполнению для анализа больших объемов данных;

– выполнение алгоритмов интеллектуального анализа в облачных средах, поддерживающих различные платформы распределенных вычислений;

– потоковая обработка разнородной, быстро изменяющейся, имеющей большие объемы информации.

Анализ существующих подходов. Очевидно, что необходима разработка новых алгоритмов для анализа больших данных. Поэтому в последнее время большое количество исследований посвящено разработке подобных алгоритмов, включая алгоритмы интеллектуального анализа для параллельной обработки данных в системах облачных вычислений. Так в работе [12] рассматривается возможность параллельной обработки данных с помощью существующих «облачных» решений, таких, как Amazon EC2, Google App Engine, Microsoft Azure, Manjras oft Aneka .

В [13] авторы анализируют несколько сервисов для параллельной обработки данных с помощью методов интеллектуального анализа в «облаке». Механизм распараллеливания, предлагаемый авторами, основан на методах Apache Hadoop парадигмы MapReduce.

В [14] авторы оценивают работу распараллеленного алгоритма Apriori на основе «облака» с установленным программным комплексом Apache Hadoop.

В [15] авторы предлагают метод распараллеливания алгоритма Naive Bayes для задач анализа данных. Для распараллеливания алгоритма также использовалась система Apache Hadoop парадигмы MapReduce. Эксперименты показали, что запуск подобного алгоритма с использованием облачной среды дал лучший результат, чем традиционный алгоритм. При этом ускорение его работы относительно традиционного алгоритма росло с увеличением количества данных, поданных на вход алгоритма.

Выделим также существующие системы, которые можно было бы отнести к облачным технологиям обработки данных. Одним из первых в этой области начал работать Китайский мобильный институт. В 2007 г. в нем начались исследования и разработки в области облачных вычислений. В 2009 г. он официально анонсировал платформу для облачных вычислений BigCloud, включающую в себя инструменты для параллельного выполнения алгоритмов Data Mining Big Cloud-Parallel Data Mining (BC-PDM) [16].

BC-PDM представляет собой SaaS-платформу, построенную на базе Apache Hadoop. Пользователи могут загружать данные в хранилище (размещенное в облаке) из разных источников и применять к ним различные приложения по управлению данными, анализу данных и бизнес-приложения. В состав приложений анализа входят параллельные приложения, выполняющие ETL-обработку, анализ социальных сетей, анализ текстов (Text Mining), анализ данных (Data Mining), статистический анализ.

В 2009 г. компания Amazon расширила свои облачные сервисы Amazon EC2 и Amazon Simple Storage Service (Amazon S3) еще одним PaaS-сервисом – Amazon Elastic Mapreduce (EMR) [17]. Данный сервис также построен на платформе Apache Hadoop и предоставляет масштабируемую инфраструктуру для выполнения созданных пользователем (по определенным правилам) приложений. Сервис позволяет загрузить в Amazon S3 необходимое приложение и/или данные, которые в дальнейшем будут выполнены на job-узлах платформы Hadoop. В состав EMR входят примеры приложений, которые могут быть загружены в сервис, в том числе и решающие задачи Data Mining. Таким образом, при анализе на EMR необходимо выполнить 4 следующих шага:

1) создать приложение для анализа данных;

2) загрузить данные и/или приложения на Amazon S3;

3) запустить job-узел через консоль управления (AWS ManagementConsole) со ссылками на ранее загруженные в Amazon S3 данные и/или приложения;

4) наблюдать через консоль управления за процессом анализа до его завершения.

Amazon EMR – первый из сервисов, который может быть классифицирован как SaaS и PaaS одновременно, в зависимости от его использования конечным пользователем.

В 2012 г. корпорация Google опубликовала свой новый облачный сервис Google BigQuer [18]. Данный сервис позволяет обрабатывать большие объемы данных, хранящихся в облаке. Если данные в облаке отсутствуют, то их предварительно необходимо загрузить туда. При загрузке данные

трансформируются во внутренние форматы (сейчас поддерживаются 2 формата CSV и JSON), в которых они потом используются для обработки. Место для загрузки данных ограничено (100 Гбайт бесплатно, все, что свыше, – платно).

Доступ к сервису может осуществляться с использованием веб-браузера, клиентского программного обеспечения, реализующего поддержку командной строки, или через API для распределенных систем, построенных в соответствии с архитектурным стилем REST. При использовании веб-браузера и консольных приложений пользователь для обработки своих данных должен ввести SQL-подобный запрос. Он может включать все те же элементы, что и Select-запрос в SQL: FROM, JOIN, WHERE и др. Таким образом, пользователь имеет возможность сформировать довольно гибкий поисковый запрос по данным произвольного типа.

Необходимо отметить, что данный сервис не решает напрямую задач Data Mining: кластеризации, классификации и др. Поэтому его нельзя рассматривать в чистом виде как Data Mining Cloud, скорее, он относится к классу OLAP-систем.

Помимо существующих открытых облачных сервисов, которые могут быть использованы для решения задач Data Mining, существует ряд разработок, на базе которых может быть построен облачный интеллектуальный анализ данных.

Корпорация Microsoft предлагает Azure Machine Learning [19] сервер, который может быть использован для решения задач Data Mining в облаке. Данный сервер построен как WCF (Windows Communication Foundation) – приложение. Он предоставляет .NET API для написания сервисно-ориентированных приложений. Кроме того он позволяет расширять состав алгоритмов добавлением соответствующих плагинов.

При доступе к данным в качестве источников данных могут использоваться текстовые файлы (CSV, TSV и с другими разделителями), файлы HDFS, таблицы Hive из Hadoop, таблицы SQL Azure, объекты и таблицы в Azure, потоки данных OData и JSON, веб-страницы.

При изучении данных можно использовать набор модулей для извлечения примеров данных (случайные, Top-N, диапазоны, расслоения), модули статистического анализа данных (распределение, корреляция, тестирование гипотез), а также очень полезна возможность визуализации наборов данных.

Для создания и выбора признаков можно использовать блоки масштабирования и функциональные преобразования, группировку цифровых характеристик, двоичное кодирование категориальных функций, выделение признаков с помощью скриптового языка R, выбор компонентов с использованием фильтров (корреляция, частота, взаимная информация, хи-квадрат) и упаковщиков (пошаговый выбор характеристик).

При разработке модели используются алгоритмы классификации (Boosted Decision Trees, Random Forests, Logistic Regression, SVM, Averaged Perceptron, Neural networks), регрессии (Linear Regression, Boosted Decision Trees, Neural networks), рекомендаций (SVD, Non-negative matrix factorization) и кластеризации (K-means).

Существуют средства для построения Data Mining «облака» и на базе популярной открытой библиотеки алгоритмов Data Mining – Weka [20]. Ее расширение Weka4WS [21] реализует распределенный каркас для поддержки выполнения алгоритмов Data Mining в среде WSRFenabled Grids. Она интегрирует библиотеку Weka и WSRF-технологию для запуска удаленных Data Mining-алгоритмов и управления распределенными вычислениями в виде Workflow. Такая интеграция позволяет на некотором удаленном вычислительном узле развернуть WSRF-совместимый сервис для публикации всех Data Mining-алгоритмов, включенных в библиотеку Weka.

В настоящее время одним из самых популярных средств Data Mining (по версии KDnuggets [22]) является система RapidMiner, разработанная одноименной компанией. Данный продукт является Open Source, и версия с минимальными функциональными возможностями распространяется бесплатно. RapidMiner реализует клиент-серверную архитектуру. Rapid Miner Server может использоваться отдельно, предоставляя возможности интеллектуального анализа в виде веб-сервисов, тем самым реализуя модель облачных вычислений – SaaS.

RapidMiner реализует все необходимые операции для анализа: загрузку и преобразование данных (ETL), предобработку данных, визуализацию данных, решение задач Data Mining. Он имеет открытую архитектуру, предоставляя возможность расширять себя новыми алгоритмами, в том числе и алгоритмами, реализованными из библиотек Weka и R.

Основными недостатками описанных систем являются:

1. Необходимость пользователю самому создавать аналитические приложения и/или SQL-подобные скрипты для их выполнения в «облаке».

2. Необходимость хранения анализируемых данных внутри облака, что в свою очередь имеет ряд недостатков:

– требует дополнительных аппаратных средств для хранения данных;

– для обработки актуальных данных нужно или всегда хранить их во внутреннем хранилище, или решить задачу их синхронизации с источником информации;

– при загрузке информации производится преобразование во внутренние форматы, что может привести к искажению информации и/или вызвать ошибку при загрузке;

– обеспечение конфиденциальности хранящейся во внутреннем хранилище информации ложится на провайдера сервиса, что не всегда может удовлетворить владельца информации.

3. Использование непараллельных и/или строго распараллеленных алгоритмов, что не позволяет гибко перестраивать алгоритмы в зависимости от выбранных параметров среды выполнения, а также вида данных.

4. Привязка только к одной технологии выполнения распределенных вычислений (в основном Map Reduce и ее реализации Apache Hadoop), каждая из которых имеет свои недостатки и может эффективно применяться только при определенных условиях.

5. Решение конечных бизнес-задач, а не отдельных задач интеллектуального анализа данных, что ограничивает возможность комбинировать данные задачи и решать более широкий круг бизнес-задач.

6. Отсутствие в большинстве решений унифицированных программных интерфейсов, предоставляющих доступ как к сервисам «облака», так и к результатам анализа.

Из приведенного обзора в области интеграции технологии интеллектуального анализа данных и облачных технологий видно, что авторы отмечают высокую актуальность данного решения, но констатируют различные существующие проблемы. Имеющиеся исследования в основном направлены на популярную в настоящее время парадигму распределенных вычислений MapReduce и ее реализацию компанией Apache Hadoop. Однако данная парадигма обладает рядом недостатков, главным из которых для алгоритмов интеллектуального анализа является необходимость наличия свойства списочного гомоморфизма [14] у распараллеливаемой функции. Однако не все алгоритмы интеллектуального анализа обладают таким свойством (например, функция вычисления информативности атрибута алгоритмов ID3, C4.5 и др.). Альтернативой могут являться другие парадигмы распределенных вычислений (например, модель акторов, предоставляющая возможность обмена промежуточными результатами за счет механизма асинхронного обмена сообщениями), но они практически не исследуются.

Полученные результаты. Авторами проведены исследования в области применения интеллектуального анализа данных к большим данным на базе различных вычислительных распределенных платформ, основными результатами которых являются:

1. Модель представления алгоритмов интеллектуального анализа данных в виде функционального выражения, состоящего из последовательности унифицированных «чистых» функций, которые могут быть выполнены параллельно, на основе теории λ-исчислений. Модель, в отличие от существующих, позволяет реструктурировать алгоритмы перестановкой и заменой блоков, а также преобразовывать к параллельной форме.

2. Библиотека параллельных алгоритмов интеллектуального анализа данных, реализованная на основе модели представления алгоритмов интеллектуального анализа данных в виде функционального выражения. В отличие от существующих библиотек она позволяет преобразовывать последовательные реализации алгоритмов к форме для параллельного выполнения без необходимости существенного изменения программного кода.

3. Архитектура и программный прототип облачной среды интеллектуального анализа данных, в отличие от существующих интегрирующие технологии интеллектуального анализа данных, распределенные вычисления и облачные технологии. Прототип включает в себя следующие компоненты:

– управления пользователями, пользовательскими проектами и их данными в облачной среде;

– настройки и управления процессом анализа данных на распределенных вычислительных системах, входящих в состав облачной среды;

– адаптеры к следующим системам параллельных и распределенных вычислений: многопоточность, AKKA (модель акторов), Apache Hadoop (MapReduce);

– системы распределенных вычислений: AKKA и Apache Hadoop.

Реализация данной стадии позволит верифицировать полученные на ней результаты, а в дальнейшем осуществлять проверку теоретических решений, создаваемых на последующих стадиях проекта.

Анализ российских и зарубежных исследований показал, что в настоящее время нет комплексных решений, обеспечивающих выполнение интеллектуального анализа больших данных в облачной вычислительной среде. Предлагаемые авторами модели и методы обладают существенной научной новизной, а построенный на их основе модельно-методический аппарат представляет собой первый подобный комплекс, предназначенный для исследования применения методов интеллектуального анализа к большим данным в облачных распределенных вычислительных средах. Таким образом, полученные методы и подходы позволяют получить результаты мирового уровня.

Модель представления алгоритмов интеллектуального анализа данных в виде функционального выражения. Для создания алгоритмов как последовательности произвольных блоков такие блоки должны обладать следующими свойствами:

– взаимозаменяемостью;

– исполнением в любом порядке.

Требование взаимозаменяемости реализуется за счет унификации входного и выходного интерфейсов блока. Все блоки для создания алгоритмов должны получать одинаковый входной набор аргументов и возвращать одинаковый набор результатов.

Возможностью выполнения в произвольном порядке обладают функции в функциональных языках программирования. Такие языки основаны на теории λ-исчислений, в рамках которой известна теорема Черча–Россера: состояния, которые могут быть достигнуты при применении редукций разным порядком к какому-либо терму в λ-исчислениях могут быть сведены к одинаковому результату.

Представим алгоритм ИАД как композицию функций и расширив λ-исчисления необходимыми встроенными функциями.

Формально алгоритм ИАД можно представить как функцию, принимающую на вход набор данных D и строящую модель знаний M [18]–[20]:

$$\mathrm{DMA}:: D \to M^1.$$

Утверждение. Алгоритм ИАД можно представить функциональным выражением в виде композиции функций:

$$\mathrm{dma} = fb_n \circ fb_{n-1} \circ \ldots \circ fb_i \circ \ldots \circ fb_1 =$$
$$= fb_n\,(d, fb_{n-1}\,(d, \ldots.\, fb_i\,(d, \ldots.\, fb_1(d, nil)\ldots)..)),$$

где fb_i – функциональный блок типа $FB:: D \to M \to M$, выполняющий некоторые вычисления (часть алгоритма) и строящий ***выходную модель знаний*** m_i на основе набора данных d и ***входной модели знаний***, построенной предыдущим функциональным блоком m_{i-1}:

$$fb_i: D \to M \to M.$$

Каждая такая функция также может быть представлена композицией функций:

$$fb_i = fb_{i.k} \circ \ldots \circ fb_{i.r} \circ \ldots fb_{i.1} =$$
$$= fb_{i.k}\,(d, \ldots.\, fb_{i.r}\,(d, \ldots.\, fb_{i.1}(d, m)\ldots)..).$$

Для унификации функций будем считать, что первая функция fb_1 (а следовательно, и алгоритм) в качестве входного аргумента получает пустую модель знаний $m_0 = \varnothing$, следовательно, алгоритм ИАД в виде функции будет описан следующим образом:

$$f: D, M \to M.$$

Унификация интерфейсов функций и использование ими для вычисления только входных аргументов (т. е. наличие у них свойств чистых функций) позволяют называть функции fb_i функциональными блоками.

Библиотека параллельных алгоритмов интеллектуального анализа данных. Для практического применения полученные результаты были реализованы в виде каркаса библиотеки алгоритмов ИАД для параллельного и распределенного выполнения. Для этого была расширена[2] библиотека Xelopes компании PrudSys (http://www.prudsys.de/). Данная библиотека, реализованная на языке программирования Java, представляет собой базовые классы для реализации алгоритмов Data Mining. В том числе она включает в себя реализации разных алгоритмов Data Mining. Ее основным недостатком является монолитная реализация алгоритмов, которая не позволяет его распараллелить без изменения кода.

[1] Здесь и далее прописными буквами будем обозначать типы, а строчными – переменные этих типов (например, переменная d типа D).

[2] Модификация библиотеки Xelopes получила название DXelopes (Distributed Xelopes). Она размещена в открытом репозитории исходных кодов BitBucket и доступна по ссылке https://bitbucket.org/iiholod/dxelopes4students.

Для устранения этого недостатка авторы модифицировали ядро данной библиотеки, представив алгоритм в виде отдельных блоков. Такое представление алгоритма соответствует подходу, описанному ранее.

В результате была разработана библиотека, обладающая следующими возможностями по сравнению с существующими библиотеками алгоритмов ИАД:

– создавать новые алгоритмы посредством комбинации существующих функциональных блоков или модификации существующих алгоритмов заменой (или модификацией) отдельных функциональных блоков;

– преобразовывать последовательные алгоритмы, построенные в виде последовательности функциональных блоков, в параллельные формы для многопотокового выполнения;

– выполнять параллельные алгоритмы ИАД на различных средствах распределенных вычислений.

Библиотека алгоритмов ИАД для параллельного и распределенного выполнения представляет собой набор взаимосвязанных классов. Они реализуют следующий базовый функционал, необходимый для выполнения алгоритмов ИАД:

– загрузку данных;

– обработку данных (доступ к различным элементам набора данных);

– настройку выполнения алгоритмов ИАД;

– структуры для хранения результатов выполнения алгоритмов ИАД – моделей знаний;

– базовые классы и типовые структуры для реализации блоков алгоритмов ИАД.

Последнее свойство отличает построенную библиотеку от существующих (в том числе и от библиотеки Xelopes, на базе которой она построена). Оно позволяет реализовать алгоритм ИАД в соответствии с ранее описанным подходом, тесно интегрировав его в библиотеку и тем самым подготовив к выполнению в параллельной и распределенной среде.

Рис. 1, *а* иллюстрирует типовой подход к построению библиотек алгоритмов ИАД, используемый в таких популярных библиотеках, как RapidMiner, Weka и Xelopes. В них новые алгоритмы добавляются как отдельные монолитные (не декомпозированные на блоки) классы. Выполнение таких алгоритмов предполагается в неизменном виде или посредством их модификации.

В отличие от них реализованная библиотека предполагает декомпозицию исходного алгоритма ИАД на функциональные блоки (рис. 1, *б*). При этом часть таких блоков может быть заимствована из состава библиотеки. Таким образом, она позволяет использовать существующий задел в библиотеке (в виде функциональных блоков) и

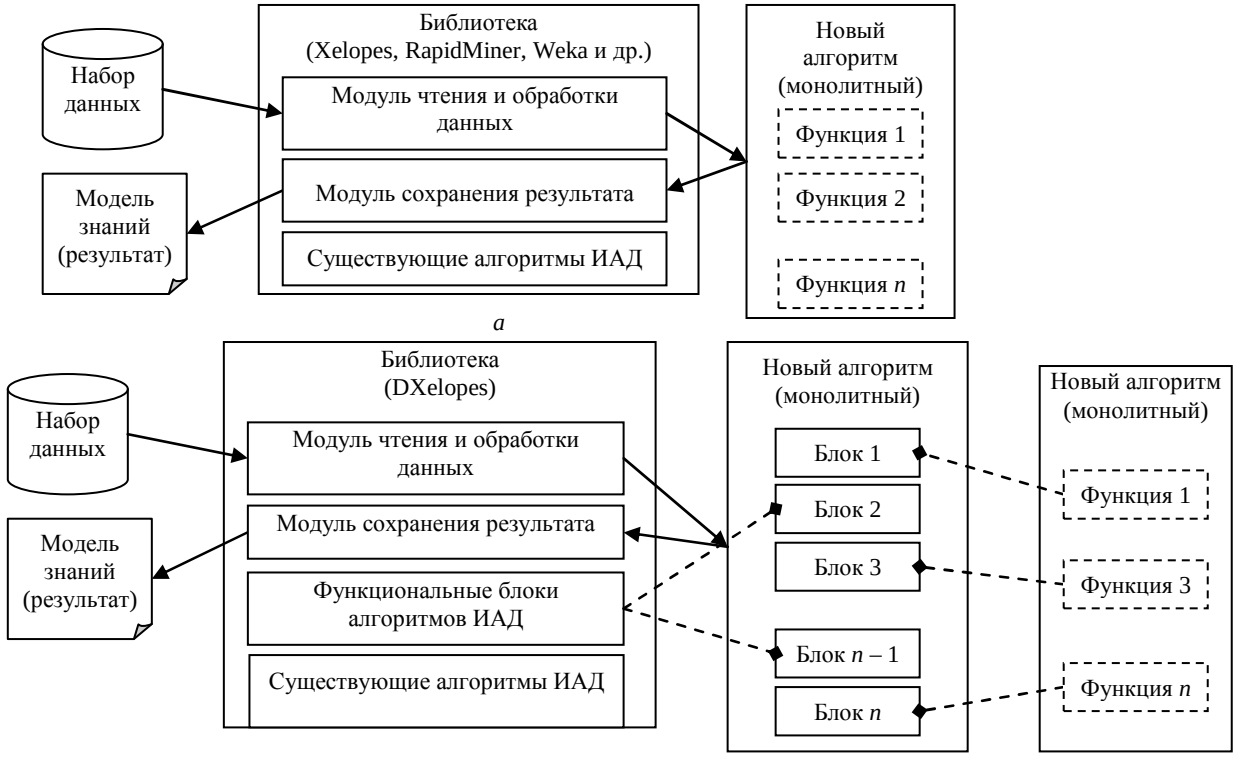

Рис. 1

реализовывать только те блоки, которые являются уникальными для алгоритма. Кроме того, такое построение алгоритма позволяет оперировать функциональным блоком как неделимой единицей, размещать и выполнять ее на отдельных вычислительных элементах.

Библиотека алгоритмов ИАД для распределенного выполнения DXelopes (рис. 2) разделена на следующие модули:

• ядро библиотеки (Core), построенное в соответствии со стандартом CWM [1] (CWM) и включающее в себя:

– средства работы с данными, заимствованные из библиотеки Xelopes;

– средства сохранения результата (модели знаний), заимствованные из библиотеки Xelopes;

– средства для настройки задач и алгоритмов ИАД, заимствованные из библиотеки Xelopes;

– базовые классы для реализации функциональных блоков в соответствии с предложенными результатами;

– среду для параллельного выполнения на многопоточных системах;

• алгоритмы классификации (Classification);

• алгоритмы кластеризации (Clustering);

• алгоритмы поиска ассоциативных правил (Association).

Последние 3 модуля используют ядро и не зависят друг от друга.

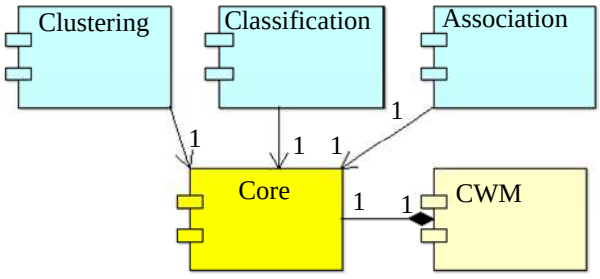

Рис. 2

Архитектура и программный прототип облачной среды интеллектуального анализа данных. Несмотря на интенсивное развитие облачных вычислений, систем интеллектуального анализа данных и обработки данных большого объема, нерешенными проблемами сегодня остаются: отсутствие библиотек параллельных алгоритмов ИАД для исследователей и разработчиков, системы распараллеливания и тестирования алгоритмов ИАД для разработчиков, online-системы ИАД для аналитиков, предоставляющих масштабируемые вычислительные ресурсы. Для устранения данной проблемы авторами была разработана платформа интеллектуального анализа данных (в том числе из разных распределенных источников) с использованием масштабируемых вычислительных ресурсов на основе облачных вычислений. Данная платформа создана в интересах:

1) разработчиков программ и алгоритмов (исследователей) в области ИАД, предоставляя среду и вычислительные мощности для отладки и исследования параллельных и распределенных версий алгоритмов, а также фреймворк и средство разработки для локальной разработки и отладки алгоритмов;

2) аналитиков, предоставляя сервисы интеллектуального анализа данных различными методами, обработки данных (преобразования, очистки и т. п.) на «неограниченных» вычислительных ресурсах.

В настоящее время реализован прототип платформы, развернутый на базе факультета компьютерных технологий и информатики Санкт-Петербургского электротехнического университета «ЛЭТИ» и используемый в учебном процессе.

Конечный целевой продукт будет представлять собой платформу для интеллектуального анализа данных на основе систем облачных вычислений при решении задач исследования и разработки различных алгоритмов ИАД. Она будет предоставлять сервисы для анализа данных, хранящихся в том числе и вне облака. Предоставляться сервисы будут как в интересах аналитиков, так и в интересах разработчиков (исследователей) в области интеллектуального анализа данных. Архитектура «облака» будет позволять расширять различные сервисы, в том числе и «частные» сервисы для использования ограниченным кругом лиц. Кроме того, она позволит распараллеливать, тестировать и оценивать характеристики разрабатываемых алгоритмов и программ ИАД, подбирать оптимальную структуру алгоритма при заданной архитектуре вычислительной среды, подбирать оптимальную архитектуру вычислительной среды для реализации алгоритмов, извлечения данных из хранилища, построения конечных моделей.

Разработанная платформа может быть портирована на аппаратные средства заказчика, тем самым создавая приватное облако интеллектуального анализа. Это обеспечит заказчику возможность использовать все преимущества платформы без необходимости передачи данных куда-либо.

Функциональная архитектура платформы представлена на рис. 3. Ядром разрабатываемой платформы является библиотека распределенных алгоритмов data mining (DXelopes). Принципы, заложенные в библиотеку (на основе функциональной модели), позволяют расширять ее новыми алгоритмами Data Mining, в том числе с использованием уже имеющихся функциональных блоков, а также выполнять их параллельно и распределенно.

Наличие в библиотеке адаптеров к разным распределенным системам позволяет интегрировать библиотеку с новыми системами распределенных вычислений, которые могут быть интегрированы в платформу. За счет этого уже имеющиеся в библиотеки алгоритмы будут работать на новых распределенных системах без необходимости их модернизации (тем самым позволяя проводить новые исследования и получать более производительный анализ без дополнительных усилий).

Наличие в библиотеке адаптеров к ETL-средствам позволяет интегрировать библиотеку с новыми средствами извлечения, обработки и загрузки данных (помимо CloverETL), которые могут быть интегрированы в платформу. Это позволяет расширить возможности платформы в части обработки данных, а пользователям, при необходимости, использовать средства, более привычные для них.

Распределенные системы представляют собой пул заранее подготовленных (имеющих предустановленное и настроенное ПО) виртуальных машин, объединенных в кластеры распределенных вычислений. В таких виртуальных кластерах установлены и настроены системы распределенных вычислений (на первой стадии предполагается интеграция с системами Java Threads, Akka (модель акторов), Apache Hadoop). Конфигурация кластера распределенных вычислений показана на рис. 4.

Практическая значимость. Практической значимостью полученных результатов являются:

– библиотека алгоритмов интеллектуального анализа для параллельного и распределенного выполнения;

– облачная среда интеллектуального анализа данных.

Библиотека алгоритмов интеллектуального анализа для параллельного и распределенного выполнения может служить основой для разработки информационно-аналитических систем в части интеллектуального анализа. При этом разрабатываемые системы могут использовать возможности параллельных и распределенных вычислений за счет параллельных реализаций алгоритмов в библиотеке. Такая возможность позволяет существенно повысить производительность алгоритмов анализа, что особенно важно для анализа больших объемов данных.

Еще одним практическим назначением библиотеки является предоставление базовых классов для разработки и отладки новых или модифицированных алгоритмов анализа. За счет блочной реализации алгоритмов и возможности формировать их посредством сборки из готовых блоков разработчик получает уникальную возможность максимально использовать имеющиеся в библиотеке наработки. Модификация существующих алгоритмов может выполняться за счет замены или модификации отдельных блоков алгоритма. Построение нового алгоритма возможно за счет разработки только уникальных блоков алгоритма.

Кроме того разработчику предоставляется возможность подобрать оптимальную структуру параллельного алгоритма перестановкой блоков и перемещением в структуре алгоритма блока распараллеливания. Таким образом он может подобрать оптимальную параллельную форму для конкретных данных и конкретной среды выполнения.

Облачная среда интеллектуального анализа данных и предоставляемые им сервисы помогут:

– решать задачи интеллектуального анализа данных из любой предметной области без дополнительного трудоемкого и дорогостоящего процесса поиска специализированных платформ и сред;

– экономить финансовые средства на создание, настройку и поддержку дорогостоящего оборудования для кластера;

– разрабатывать, тестировать, отлаживать и применять для получения окончательного результата исследований собственные алгоритмы в единой среде и на единой платформе;

– обеспечивать пользователя богатым выбором методов и средств анализа данных на масштабируемых вычислительных ресурсах без необходимости открывать доступ к своим данным;

– интегрировать усилия исследователей в области интеллектуального анализа данных, предоставляя им полигон для исследований и аналитиков, предоставляя им доступ к последним достижениям исследователей в этой области.

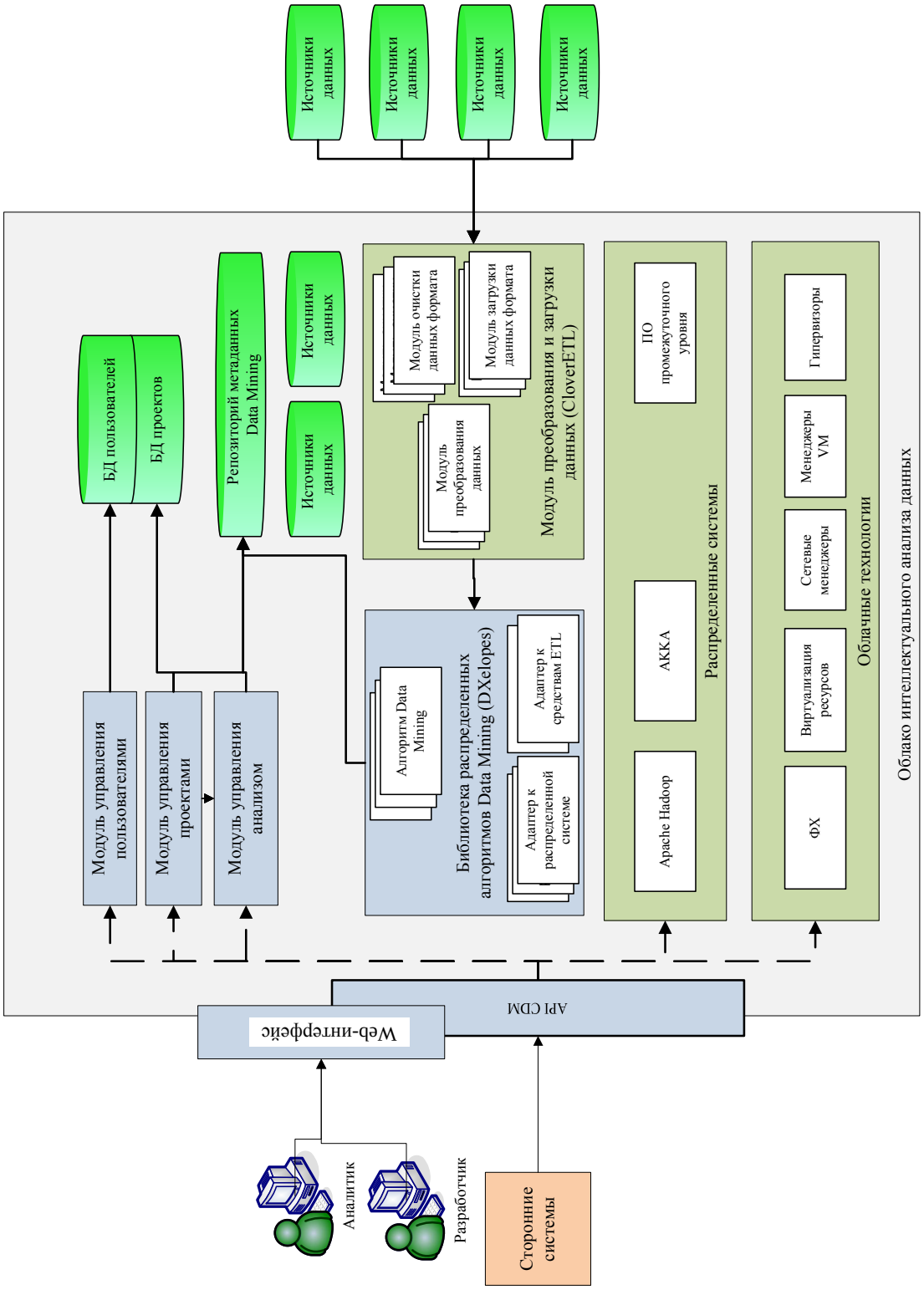
Облако интеллектуального анализа данных
Облачные технологии
ФХ
Виртуализация ресурсов
Сетевые менеджеры
Менеджеры VM
Гипервизоры
Распределенные системы
Apache Hadoop
AKKA
ПО промежуточного уровня
Библиотека распределенных алгоритмов Data Mining (DXelopes)
Адаптер к распределенной системе
Адаптер к средствам ETL
Алгоритм Data Mining
Модуль преобразования и загрузки данных (CloverETL)
Модуль преобразования данных
Модуль загрузки данных формата
Модуль очистки данных формата
Модуль управления анализом
Модуль управления проектами
Модуль управления пользователями
Репозиторий метаданных Data Mining
БД проектов
БД пользователей
Источники данных
Источники данных
Источники данных
Источники данных
Источники данных
Источники данных
API CDM
Web-интерфейс
Аналитик
Разработчик
Сторонние системы

Рис. 3

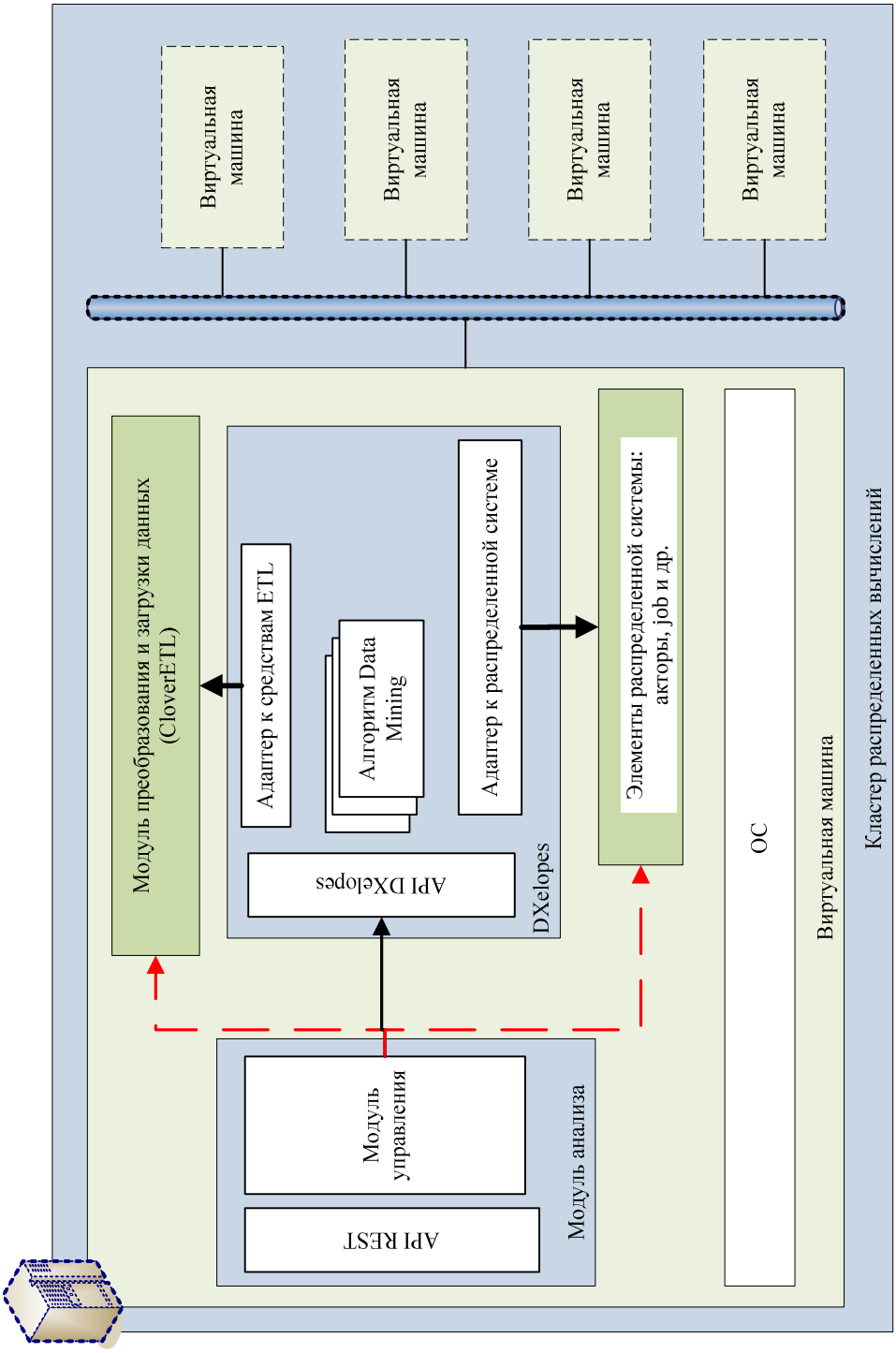
Виртуальная машина
Виртуальная машина
Виртуальная машина
Виртуальная машина
Модуль преобразования и загрузки данных (CloverETL)
Адаптер к средствам ETL
Алгоритм Data Mining
Адаптер к распределенной системе
API DXelopes
DXelopes
Элементы распределенной системы: акторы, job и др.
ОС
Виртуальная машина
Кластер распределенных вычислений
Модуль управления
API REST
Модуль анализа

Рис. 4

Перспективы развития. В перспективе планируются исследования в следующих направлениях:

1. Для выполнения алгоритмов интеллектуального анализа больших данных в облачных средах на распределенных вычислительных системах будут разработаны методы отображения параллельных алгоритмов на различные парадигмы распределенных вычислений (модель акторов, MapReduce и др.). Данные методы позволят, без изменения алгоритма, размещать его на распределенных средствах для дальнейшего выполнения.

В настоящее время не существует методов, позволяющих выполнять алгоритмы интеллектуального анализа на разных средствах распределенных вычислений без существенной реструктуризации самих алгоритмов. Проводятся исследования по выполнению алгоритмов на отдельных средствах, реализующих определенные парадигмы (MapReduce, сервисно-ориентированные архитектуры, многоагентные системы). Однако большинство из них предполагают изменение структуры параллельного алгоритма под определенную структуру (например, явное выделение функций map и reduce для парадигмы MapReduce). Кроме того, ни одно из них не предполагает универсального подхода для разных парадигм распределенного вычисления.

Данные методы будут основаны:

– на ранее разработанной модели представления алгоритмов интеллектуального анализа данных, использующей элементы теории λ-исчислений и представляющие алгоритмы в виде функционального выражения, состоящего из последовательности унифицированных «чистых» функций, которые могут быть выполнены параллельно;

– на разработанной в рамках проекта модели распределенных вычислений в облачной среде, позволяющей описать различные парадигмы, основанной на модели акторов.

Учитывая соответствие между λ-исчислениями и моделью акторов, указанные модели будут иметь прямое отображение друг на друга, что позволит с их помощью разработать методы размещения алгоритмов интеллектуального анализа данных на средствах распределенного вычисления.

Наличие реализации модели алгоритмов в виде библиотеки алгоритмов и реализации парадигм распределенных вычислений позволит выполнить программную реализацию методов. При этом программные адаптеры, являющиеся основой интеграции библиотеки и распределенных средств, будут создаваться на основе построенной модели распределенных вычислений.

2. Для интеграции алгоритмов интеллектуального анализа с большими данными будут разработаны методы интеграции и преобразования данных из разных источников и адаптации их к применению алгоритмов интеллектуального анализа. Данные методы будут реализовывать:

– ETL процесс, что в свою очередь позволит использовать существующие ETL-средства и имеющиеся у них средства преобразования данных;

– спецификацию CWM (в частности, пакет Transformation), что позволит интегрировать предложенный метод в существующую библиотеку параллельных алгоритмов интеллектуального анализа данных, построенную в соответствии с данной спецификацией;

– потоковую обработку данных, что позволит выполнять анализ данных «на лету» без необходимости их загрузки в облачную среду. Это в свою очередь снизит требования к дисковому пространству «облака» и безопасности, причем потери в производительности будут компенсироваться за счет кеширования.

Разработанные методы в отличие от существующих будут позволять алгоритму «управлять» процессом получения данных (соответственно их выгрузки, очистки и преобразования). Это в свою очередь позволит обращаться только к данным, необходимым для анализа, без их сохранения в облачной среде. При этом возможность кеширования данных сохранится.

3. Для интеграции технологий интеллектуального анализа данных (data mining), параллелизации алгоритмов, выполнения распределенных вычислений, выгрузки, преобразования и загрузки данных (ETL-технологии), обработки больших данных, облачных вычислений в единую технологию облачного интеллектуального анализа больших данных будет предложен подход к построению облачной вычислительной среды обработки и анализа больших данных. Данный подход в отличие от существующих объединит 3 модели облачных вычислений:

– SaaS (Software as a Service) – обеспечивающую конечному пользователю (аналитику) доступ к уже имеющимся в облачной среде алгоритмам анализа, сервисам по обработке данных (загрузке, интеграции, очистки и др.) и полученным результатам анализа;

– PaaS (Platform as a Service) – обеспечивающую разработчикам и исследователям в области анализа больших данных платформу для размещения своих алгоритмов анализа и обработки данных и проведения необходимых исследований;

– IaaS (Infrastructure as a Service) – обеспечивающую доступ к масштабируемым вычислительным ресурсам, на которых с использованием разных средств распределенных вычислений могут выполняться алгоритмы анализа и обработки больших данных.

Проведенный анализ существующих облачных сред в области обработки больших данных показал, что на данный момент существуют системы, реализующие только 2 уровня (например, Amazon EMR реализует SaaS- и PaaS-уровни). Таким образом, предлагаемая модель и последующая ее реализация не имеют аналогов в мире.

Интеграционной основой, объединяющей перечисленные технологии, будет функциональное представление алгоритмов, предполагающее применение принципов функциональной парадигмы программирования и имеющей под собой теоретическую основу в виде теории λ-исчислений, разработанной Алонзо Чёрчем.

Разрабатываемый подход будет использовать объектную модель сущностей, порождаемых в процессе выполнения интеллектуального анализа больших данных, которая будет охватывать весь цикл управления процессом анализа. Это позволит построить облачную вычислительную среду интеллектуального анализа данных не только в интересах исследователей методов анализа, но и в интересах конечных пользователей, выполняющих такой анализ. Как следствие, разработанный подход может найти практическое применение в построении как публичных, так и приватных облачных сред в научных, промышленных и коммерческих целях.

СПИСОК ЛИТЕРАТУРЫ

1. Методы и модели анализа данных: OLQP и Data Mining / А. А. Барсегян, М. С. Куприянов, В. В. Степаненко, И. И. Холод. СПб.: БХВ-Петербург, 2004.

2. Методы анализа данных: Data Mining, Visual Mining, Text Mining / А. А. Барсегян, М. С. Куприянов, В. В. Степаненко, И. И. Холод // OLAP. 2-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2007.

3. Анализ данных и процессов/ М. С. Куприянов, А. А. Барсегян, И. И. Холод, С. И. Елизаров. СПб.: БХВ-Петербург, 2009.

4. Куприянов М. С., Холод И. И. Управление требованиями в программных проектах. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2011.

5. Распределенные системы анализа данных на базе облачных вычислений / М. С. Куприянов, И. А. Голубев, И. И. Холод и др. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2011.

6. Интеллектуальный анализ распределенных данных на базе облачных вычислений / М. С. Куприянов, И. А. Голубев, И. И. Холод и др. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2011.

7. Analysis od data and processes: from standart to realtime data mining / A. Barsegian, M. Kupriyanov, I. Holod, S. Elizarov, M. Tess // ReDi Roma-Verlag. 2014.

8. Gartner Survey Reveals That 73 Percent of Organizations Have Invested or Plan to Invest in Big Data in the Next Two Years. Press Release. Gartner. STAMFORD, Conn., September 17, 2014. URL: http://www.gartner.com/newsroom/id/2848718.

9. Smets-Solanes J.-P., Carvalho R. A. Cloud computing and SaaS: New data mining tools for the IRS. URL: http://www.emqus.com/index.php?/emq/article/cloud_computing_and_saas_new_data_mining_tools_for_the_irs_998.

10. Big Data Has Exhaust Problem. URL: http://www.informationweek.com/big-data/big-data-analytics/big-data-has-exhaust-problem/d/d-id/1278765.

11. Marcus G., Davis E. Eight (No, Nine!) Problems With Big Data. URL: http://www.nytimes.com/2014/04/07/opinion/eight-no-nine-problems-with-big-data.html?_r=0.

12. Patil V., Nikam V. B. Study of Data Mining algorithm in cloud computing using MapReduce Framework. URL: http://borjournals.com/Research_papers/Jul_2013/1390IT.pdf.

13. Geng X., Yang Z. Data Mining in Cloud Computing. URL: https://www. google. com/.

14. Lijuan Zhou, Hui Wang, Wenbo Wang. Parallel Implementation of Classification Algorithms Based on Cloud Computing Environment // TELKOMNIKA Indonesian J. of Electrical Engineering. 2012. Vol. 10, № 5. P. 1087–1092.

15. BC-PDM: Data mining, social network analysis and text mining system based on cloud computing / L. Yu, J. Zheng, W. C. Shen, B. Wu, B. Wang, L. Qian, B. R. Zhang // Proc. of the 18th ACM SIGKDD Intern. conf. on Knowledge discovery and data mining, New York, 2012. P. 1496–1499.

16. Amazon Elastic MapReduce. Developer Guide. URL: http://s3.amazonaws.com/awsdocs/ElasticMapReduce/latest/emr-dg.pdf.

17. Google Developers. Google BigQuery. URL: https://developers.google.com/bigquery/what-is-bigquery?hl=ru.

18. Gronlund C. J. Introduction to machine learning on Microsoft Azure. URL: http://azure.microsoft.com/en-gb/documentation/articles/machine-learning-what-is-machine-learning/.

19. Weka: Practical machine learning tools and techniques with Java implementations. (Working paper 99/11) / I. H. Witten, E. Frank, L. Trigg et al. Hamilton, New Zealand: University of Waikato, Department of Computer Science, 1999.

20. Talia D., Trunfio P., Verta O. The Weka4WS framework for distributed data mining in service-Oriented Grids, Concurrency and Computation // Practice and Experience. 2008. № 20. P. 1933–1951.

21. KDnuggets Annual Software Poll:RapidMiner and R vie for first place, KDnuggets, June 2013. URL: http://www.kdnuggets.com/2013/06/kdnuggets-annual-software-poll-rapidminer-r-vie-for-first-place.html.

22. David Norris, RapidMiner – a potential game changer, IT-Director.com, November 22, 2013. URL: http://www.it-director.com/content.php?cid=14555.

M. S. Kupriyanov, I. I. Holod, A. V. Shorov, Yu. A. Shichkina
Saint Petersburg Electrotechnical University «LETI»

INFORMATION SYSTEMS INTELLIGENT DATA ANALYSIS AND PROCESSES (PROBLEM BIG DATA) THE REALIZATION OF NAIVE

The article describes the approach of building a cloud platform for data mining. The approach of decomposition algorithms into functional blocks, allowing to distribute their performance on distributed nodes. The description of the architecture of the platform and place it in the library of data mining algorithms that implements considered approach.

Data mining, cloud, big data