

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.9

BIG DATA: ПРОБЛЕМЫ, МЕТОДЫ АНАЛИЗА, АЛГОРИТМЫ

*МАГЕРРАМОВ З.Т., АБДУЛЛАЕВ В.Г.,
МАГЕРРАМОВА А.З.*

Проводится обзор развития, характеристики и применения технологий Big Data, показывается взаимосвязь технологии Big Data с промышленными предприятиями на примере металлургического производства. Описываются принципы обработки Big Data на основе модели Apache Hadoop и компании Oracle, а также методы анализа массивов данных. Некоторые методы анализа данных сопровождаются алгоритмами кластеризации и в них применяется функция конкурентного сходства (FRiS-функция).

Ключевые слова: большие массивы данных, металлургическое производство, обработка и методы анализа, многомерный анализ, регрессия, классификация, кластеризация, алгоритмы кластеризации, FRiS-функция.

1. Введение

Термин Big Data относится к наборам данных, размер которых превосходит возможности типичных баз данных по хранению, управлению и анализу информации. Сами по себе алгоритмы Big Data возникли при внедрении первых высокопроизводительных серверов (мэйнфреймов), обладающих достаточными ресурсами для оперативной обработки информации и пригодных для компьютерных вычислений и дальнейшего анализа.

Предпринимателей и ученых волнуют вопросы качественной интерпретации данных, разработки инструментов для работы с ними и развития технологий хранения. Этому способствует внедрение и активное использование облачных моделей хранения и вычислений.

За последние три года человечество произвело информации больше, чем за всю историю своего существования до 2008 года. И рост продолжается экспоненциально. В настоящее время множество компаний следят за развитием технологий Big Data. В отчете, представленном компанией IDC [1] в декабре 2012 г., предсказывалось, что объемы информации будут удваиваться каждые 2 года в течение следующих 8 лет. За ближайшие 7 лет количество данных в мире достигнет 40 зеттабайт ($1 \text{ ЗБ} = 10^{21}$ байт), а это значит, что на каждого жителя Земли будет приходиться по 5200 ГБ данных.

В настоящее время большинство крупнейших поставщиков информационных технологий для организаций в своих деловых стратегиях используют понятие «большие данные», а основные ана-

литики рынка информационных технологий посвящают концепции выделенные исследования. Если обратить внимание на динамику роста данных, то обнаружим рост вычислительных средств, приложений и пользователей – от миллионов в эпоху мэйнфреймов до сотен миллионов в эпоху ПК и миллиардов пользователей в эпоху мобильных устройств, мобильного Интернета, социальных сетей, «облачных» технологий и построения всевозможных решений «умной» экономики. Отдельная тема – Интернет и мобильные технологии. Поток информации генерируется всё новыми интернет-сервисами, социальными сетями, приложениями электронной торговли, приложениями о местонахождении абонентов сетей. Количество e-mail, отправляемых каждую секунду в мире – 2,9 млн, объем видео, загружаемого на YouTube каждую минуту – 20 часов. Объем данных, обрабатываемых Google за день – 24 петабайт. Количество сообщений на Твиттере в день – 50 млн. Количество минут, проведенных в Facebook в месяц – около 700 млрд. Объем данных, переданных/полученных на мобильные устройства – 1,3 экзбайт. Количество продуктов, заказываемых в Amazon в секунду – 72,9 [2].

2. Постановка задачи

Методику и инструменты работы со структурированными данными ИТ-индустрия создала давно – это реляционная модель данных и системы управления БД. Но современной тенденцией является потребность обработки большого объема неструктурированных данных, и это та область, где прежние подходы работают плохо или вообще не работают. Именно эта потребность требует новой методики обращения с данными, и сейчас все более популярной становится модель работы с Big Data. Занимаясь проблемами Big Data, перед разработчиками и учеными стоит задача – найти программное и техническое решение, способное легко интегрироваться в существующую инфраструктуру ЦОД и обеспечить все три этапа обработки информации: сбор, ее организацию и анализ.

3. Этапы и методы решения

3.1. Характеристики технологии Big Data. В качестве характеристик для больших данных Forrester определяет понятие Big Data как технологию в области аппаратного и программного обеспечения, которая объединяет, организует, управляет и анализирует данные, характеризующиеся «четырьмя V»: объемом (Volume), разнообразием (Variety), изменчивостью (Variability) и скоростью (Velocity) [2].

Эти характеристики являются существенными проблемами технологии Big Data. Рассмотрим каждую из этих составляющих.



Объем накопленных данных в корпорациях из разных сфер деятельности (источник: McKinsey)

Объем данных (Volume). Во введении мы описали лавинообразный рост объема данных в научных и персональных приложениях. Дополним картину информацией о том, какие объемы данных накопили корпорации. Только в США это более 100 Тбайт данных. При этом в разных вертикальных индустриях объем данных существенно различается, следовательно, актуальность применения технологии Big Data в них различна (рисунок) [2].

Разнообразие форматов данных (Variety). Способность приложения обрабатывать большие массивы данных, поступающих из разных источников в различных форматах, является главным критерием отнесения его к технологии Big Data. Многие бизнес-задачи и научные эксперименты требуют совместной обработки данных различных форматов – это могут быть табличные данные в СУБД, иерархические данные, текстовые документы, видео, изображения, аудиофайлы. Пример подобного рода задачи из области медицины: как найти оптимальный курс лечения для конкретного пациента, базируясь на огромном количестве историй болезней пациентов (которые постоянно меняются), а также на базе данных медицинских исследований и геномных данных? Другой пример – из области оптимизации бизнес-процессов: как провести анализ структурированных данных из ERP (*Enterprise Resource Planning*) – приложения, а также слабоструктурированных данных в виде логфайлов и неструктурированного текста из отзывов покупателей? Третий пример – из сферы прогнозирования погоды: как выполнить анализ климата на базе многолетних метеорологических данных и данных, поступающих со спутника в реальном времени?

Скорость поступления и обработки информации (velocity). В области Big Data выделяют ещё одну проблему: недостаточно высокая скорость обработки данных. В ряде задач эта скорость должна быть очень высокой. Например, биржевым игрокам иногда нужно мгновенно принять решение, основываясь на большом количестве данных о состоянии рынка — за пару секунд ситуация уже может измениться. Очень большая скорость поступления данных характерна для многих научных задач. Например, только один экспериментальный синхротрон Advanced Photon Source (APS) в Аргоннской лаборатории, используемый в числе прочего для томографической съемки объектов на субмикронных разрешениях, может ежедневно генерировать 150 Тбайт информации.

Ценность для бизнеса (Value). Компания IDC тоже выделяет «четыре V», характеризующие данные, однако параметр Variety (изменчивость), который применяет компания Forrester, она заменяет на параметр Value (ценность). IDC подчеркивает, что параметр Value — один из основных, позволяющих выделить Big Data как новое явление. Он относится к экономическому эффекту, который технология Big Data обеспечивает пользователям. Информация – это главный аспект успешного прогнозирования роста и составления маркетинговой стратегии в умелых руках маркетолога. Big Data является точнейшим инструментом маркетолога для предсказания будущего компании.

Применение технологии Big Data может быть полезно для решения следующих задач:

- лучше узнавать своих потребителей, привлекать аналогичную аудиторию в Интернете;
- оценивать уровень удовлетворенности клиентов;
- находить и внедрять новые способы, увеличивающие доверие клиентов;
- создавать проекты, пользующиеся спросом по прогнозированию рыночной ситуации;
- маркетинг и оптимизация продаж;
- эффективно сегментировать клиентов;
- совершенствовать качество товаров и услуг;
- принимать более обоснованные управленческие решения на основе анализа Big Data;
- оптимизировать портфель инвестиций;
- повышать производительности труда.

3.2. Big Data и промышленные предприятия.

Сфера использования технологий Big Data обширна. Известно, что основной поток информации генерируют не люди. Источником служат роботы, находящиеся в постоянном взаимодействии друг с другом. Это приборы для мониторинга,

сенсоры, системы наблюдения, операционные системы персональных устройств, смартфоны, интеллектуальные системы, датчики и прочее. Все они задают бешеный темп роста объема данных, что приводит к появлению потребности наращивать количество рабочих серверов (и реальных, и виртуальных) – как следствие, расширять и внедрять новые data-центры. На промышленных предприятиях объем информации неуклонно растет за счет данных, полученных с датчиков, измерительных и «умных» устройств. Самыми перспективными устройствами считаются датчики, которые могут передавать данные в режиме реального времени. Все устройства на предприятии с помощью таких датчиков могут быть объединены в сеть, а технологии Big Data позволят обрабатывать информацию, поступающую с них, и проводить необходимые мероприятия в автоматическом режиме. Например, предприятия могут с помощью датчиков получать ежеминутные данные о состоянии своего оборудования и на основе этих данных предсказывать оптимальное время для замены и обслуживания. Слишком ранняя замена приведет к дополнительным расходам, а поздняя – к потере прибыли вследствие простоя оборудования.

Рассмотрим взаимосвязь технологии Big Data с промышленными предприятиями на примере металлургического производства. В настоящее время металлургические компании стремятся к сокращению издержек, что позволит им быть конкурентоспособными на металлургическом рынке. Использование технологии Big Data может дать значительный экономический эффект в виде сокращения затрат на обработку информации в производстве, логистике и управлении.

Современное металлургическое производство представляет собой сложный комплекс различных переделов, базирующийся на месторождениях руд, коксующихся углей, энергетических мощностях.

Металлургическое производство включает в себя следующие комбинаты, заводы, цеха:

- шахты и карьеры по добыче руд и каменных углей;
- горно-обогатительные комбинаты для подготовки руды к плавке;
- коксохимические заводы или цеха для подготовки углей к коксованию и извлечению полезных химических продуктов;
- энергетические цеха для получения сжатого воздуха, кислорода, а также очистки газов металлургических производств;
- доменные цеха для выплавки чугуна и ферросплавов;
- заводы для производства ферросплавов;

– сталеплавильные цеха (конвертерные, мартеновские, электросталеплавильные) для производства стали;

– прокатные цеха для получения сортового проката (листы, балки, рельсы, прутки, проволока) [4].

Автоматизированные системы управления металлургическими комбинатами ежесекундно порождают данные о процессах:

- технологических (АСУ ТП);
- логистических (АСУ транспортной логистики);
- управления (MES – Manufacturing execution system и ERP – Enterprise Resource Planning системы).

Системы АСУ ТП собирают данные с датчиков агрегатов о состоянии и режимах технологических процессов. С систем контроля качества могут поступать видеоизображения полос прокатки и дефектов на полосе, карты ультразвукового контроля. АСУ транспортной логистики содержат данные о перемещении материалов. ERP и MES владеют информацией о заказах, планировании, оперативном управлении обработкой материалов, о состоянии запасов на складах.

Например, только цепочка производства от выплавки металла до выпуска автолиста может включать в себя от 7000 до 15000 источников разнородных неструктурированных данных, поступающих в реальном масштабе времени. Высокая степень автоматизации производства порождает у персонала предприятий «иллюзию доступности данных» [5].

Оснащение производства современными системами автоматизации приводит к оцифровке всех получаемых данных, и это создает у персонала предприятия иллюзию их доступности. Но «оцифровано» – не значит «доступно».

Данные о технологических процессах есть в АСУ ТП агрегатов, данные о производстве – в MES системах, данные о заказах – в ERP.

Перечисленные выше характеристики Big Data («четыре V») хорошо подходят для структурированных и неструктурированных данных металлургического производства:

- множество сигналов с датчиков контроля технологических процессов,
- карты ультразвукового контроля,
- изображения полос прокатки, содержащих дефекты на полосе,
- данные о перемещении продукции и материалов,
- данные о заказах и поставщиках.

Технология Big Data позволит свести данные из АСУ ТП, АСУ транспортной логистики и систем класса ERP и MES воедино, тратя на это в разы меньшее количество времени, по сравнению с

традиционным подходом. Экономия времени, соответственно, принесет экономический эффект, делая системы обработки Big Data выгодным делом для металлургического бизнеса в целом.

В перспективе металлургические компании, благодаря Big Data, получают возможность заниматься предиктивным анализом, к примеру, с большой вероятностью предсказывать долю брака в металлургическом переделе на основе технологии машинного обучения.

3.3. Обработка и методы анализа Big Data. С точки зрения обработки в основу технологий Big Data положены два основных принципа:

- 1) распределенного хранения данных;
- 2) распределенной обработки, с учетом локальности данных.

Распределенное хранение решает проблему большого объема данных, позволяя организовывать хранилище из произвольного числа отдельных простых носителей. Хранение может быть организовано с разной степенью избыточности, обеспечивая устойчивость к сбоям отдельных носителей.

Распределенная обработка с учетом локальности данных означает, что программа обработки доставляется на вычислитель, находящийся как можно ближе к обрабатываемым данным. Это принципиально отличается от традиционного подхода, когда вычислительные мощности и подсистема хранения разделены и данные должны быть доставлены на вычислитель. Таким образом, технологии Big Data опираются на вычислительные кластеры из множества вычислителей, снабженных локальной подсистемой хранения.

Доступ к данным и их обработка осуществляются специальным программным обеспечением. Наиболее известным и интенсивно развиваемым проектом в области Big Data является Apache Hadoop [6,7]. В настоящее время на рынке информационных систем и программного обеспечения синонимом Big Data является технология Hadoop, которая представляет собой программный фреймворк, позволяющий хранить и обрабатывать данные с помощью компьютерных кластеров, используя парадигму MapReduce.

Основными составляющими платформы Hadoop являются:

- отказоустойчивая распределенная файловая система Hadoop Distributed File System (HDFS), при помощи которой осуществляется хранение;
- программный интерфейс Map Reduce, который является основой для написания приложений, обрабатывающих большие объемы структурированных и неструктурированных данных параллельно на кластере, состоящем из тысяч машин;
- Apache Hadoop YARN, выполняющий функцию управления данными.

В соответствии с подходом MapReduce обработка данных состоит из двух шагов: Map и Reduce. На шаге Map выполняется предварительная обработка данных, которая осуществляется параллельно на различных узлах кластера. На шаге Reduce происходит сведение предварительно обработанных данных в единый результат.

В основе модели работы Apache Hadoop лежат три основных принципа. Во-первых, данные равномерно распределяются на внутренних дисках множества серверов, объединенных HDFS. Во-вторых, не данные передаются программе обработки, а программа – к данным. Третий принцип – данные обрабатываются параллельно, причем эта возможность заложена архитектурно в программном интерфейсе Map Reduce. Таким образом, вместо привычной концепции «база данных+сервер» у нас имеется кластер из множества недорогих узлов, каждый из которых является и хранилищем, и обработчиком данных, а само понятие «база данных» отсутствует. Платформа Hadoop позволяет сократить время на обработку и подготовку данных, расширяет возможности по анализу, позволяет оперировать новой информацией и неструктурированными данными.

Компания Oracle разбивает жизненный цикл обработки информации на три этапа и использует для каждого из них собственное решение:

1) Сбор, обработка и структурирование данных. В качестве решения применяется Oracle Big Data Appliance – это предустановленный Hadoop-кластер, Oracle NoSQL Database и средства интеграции с другими хранилищами данных. Задача Oracle Big Data Appliance состоит в хранении и первичной обработке неструктурированной или частично структурированной информации, т.е. как раз в том, что у систем на базе Hadoop получается лучше всего.

2) Агрегация и анализ данных. Для работы со структурированными данными используется комплекс Oracle Exadata. Модули интеграции Oracle Big Data Appliance позволяют оперативно загружать данные в Oracle Exadata, а также получать доступ к данным «на лету» из Oracle Exadata.

3) Аналитика данных в реальном времени. Для максимально оперативного анализа полученных данных используется Oracle Exalytics Database Machine, которая позволяет решать аналитические задачи фактически в режиме «online».

Существует множество разнообразных методов анализа массивов данных, в основе которых лежит примерно одинаковый набор инструментов анализа данных [3]: многомерный анализ (OLAP), регрессия, классификация, кластеризация и поиск закономерностей. Некоторые из перечисленных методик вовсе не обязательно применимы исключительно к большим данным и могут с успехом

использоваться для меньших по объему массивов (например, А/В-тестирование, регрессионный анализ).

Многомерный анализ — суть метода заключается в построении многомерного куба и получении его различных срезов. Результатом анализа, как правило, является таблица, в ячейках которой содержатся агрегированные показатели (количество, среднее, минимальное или максимальное значение и так далее). В зависимости от реализации, существуют три типа системы многомерного анализа (OLAP): многомерная OLAP (Multidimensional OLAP – MOLAP); реляционная OLAP (Relational OLAP – ROLAP); гибридная OLAP (Hybrid OLAP – HOLAP).

Среди них ROLAP-системы являются наиболее прозрачными и изученными, поскольку основываются на широко распространенных реляционных СУБД, в то время как внутреннее устройство MOLAP и HOLAP обычно более закрыто и относится к области «ноу-хау» конкретных коммерческих продуктов.

MOLAP представляет информацию в виде «честной» многомерной модели, но внутри используются те же подходы, что и в ROLAP: схемы «звезда» и «снежинка». С точки зрения СУБД база данных ROLAP — это обыкновенная реляционная база, и для нее необходимо поддерживать весь перечень операций. Однако это не позволяет, во-первых, жестко контролировать этапы ввода данных. Во-вторых, собирать статистику и подбирать оптимальные структуры для хранения индексов. В-третьих, оптимизировать размещение данных на диске для обеспечения высокой скорости ввода/вывода. В-четвертых, при выполнении аналитических запросов из-за высоких требований к быстродействию нет возможности произвести глубокий статистический анализ и выработать оптимальный план выполнения. В ROLAP используются «родные» реляционные оптимизаторы запроса, которые никак не учитывают «многомерность» базы данных. Технологии MOLAP лишены перечисленных недостатков и благодаря этому позволяют добиться большей скорости анализа.

Выбор технологии MOLAP/ROLAP/HOLAP при анализе Big Data зависит от частоты обновления базы данных. С точки зрения распараллеливания обработки, на первый взгляд, все просто — любой многомерный куб может быть «разрезан» по делениям одного из измерений и распределен между несколькими серверами. Например, можно разделить куб на временные периоды (по годам и месяцам), по территориальному признаку (каждый сервер отвечает за свой регион) и так далее. Критерием для деления куба является следующий принцип: выполнение многомерного запроса

должно ложиться не на один сервер, а на несколько, после чего полученные результаты собираются в единое целое. Например, если пользователь запрашивает статистику продаж по стране за указанный промежуток времени, а данные распределены по нескольким региональным OLAP-серверам, то каждый сервер возвращает свой собственный ответ, которые затем собираются воедино. Если же данные будут распределены по временному критерию, то при выполнении рассматриваемого примера запроса вся нагрузка ляжет на один сервер.

Проблема в том, что, во-первых, очень трудно заранее определить оптимальное распределение данных по серверам, а во-вторых, для части аналитических запросов может быть заранее неизвестно, какие данные и с каких серверов понадобятся.

Применительно к Большим Данным это означает, что существующие подходы для многомерного анализа могут хорошо масштабироваться и что они допускают распределенный сбор информации — каждый сервер может самостоятельно собирать информацию, осуществлять ее очистку и загрузку в локальную базу.

Регрессия — под регрессией понимают построение параметрической функции, описывающей изменение указанной числовой величины в указанный промежуток времени. Эта функция строится на основе известных данных, а затем используется для предсказания дальнейших значений этой же величины. На вход метода поступает последовательность пар вида «время — значение», описывающая поведение этой величины при заданных условиях, например, количество продаж конкретного вида товара в конкретном регионе. На выходе — параметры функции, описывающей поведение исследуемой величины.

Независимо от вида используемой параметрической функции подбор значений ее параметров осуществляется одним и тем же способом. Вычисляется суммарная разница между наблюдаемыми значениями и значениями, которые дает функция при текущих значениях ее параметров. Затем определяется, как следует подкорректировать значения параметров для того, чтобы уменьшить текущую суммарную разницу. Эти операции повторяются до тех пор, пока суммарная разница не достигнет необходимого минимума или ее дальнейшее уменьшение станет невозможным.

С точки зрения обработки данных при регрессионном анализе ключевыми операциями являются вычисление текущей суммарной разницы и корректировка значений параметров. Если первая операция распараллеливается очевидным образом (сумма вычисляется по частям на отдельных серверах, а затем суммируется на центральном

сервере), то со второй сложнее. В наиболее общем случае при корректировке весов используют общеизвестный математический факт: функция нескольких параметров возрастает в направлении градиента и убывает в направлении, обратном градиенту. В свою очередь, вычисление градиента состоит в вычислении частных производных функции по каждому из параметров, что сводится к дискретному дифференцированию, основанному на вычислении взвешенных сумм. В результате корректировка значений параметров также сводится к суммированию, которое может быть распараллелено.

Если регрессионный анализ сводится к вычислению взвешенных сумм, то он обладает примерно той же степенью применимости и при работе с Big Data, что и многомерный анализ. Таким образом, системы регрессионного анализа вполне могут масштабироваться и работать в условиях распределенного сбора информации.

Классификация – ее задача отчасти похожа на задачу регрессии и заключается в попытке построения и использования зависимости одной переменной от нескольких других. Например, имея базу данных о цене объектов недвижимости, можно построить систему правил, позволяющую на основе параметров нового объекта предсказать его примерную цену. Отличие классификации от регрессии состоит в том, что анализируется не временной ряд – подаваемые на вход значения никак не могут быть упорядочены.

На текущий момент разработано множество методов классификации (функции Байеса, нейронные сети, машины поддерживающих векторов, деревья решений и т. д.), каждый из которых имеет под собой хорошо проработанную научную теорию. Вместе с тем все методы классификации строятся по одной и той же схеме. Сначала производится обучение алгоритма на сравнительно небольшой выборке, а затем – применение полученных правил к остальной выборке. На первом этапе возможно копирование массива данных на один сервер для запуска «классического» алгоритма обучения без распараллеливания работы. Однако на втором этапе данные могут обрабатываться независимо — система правил, полученная по итогам самообучения, копируется на каждый сервер, и через нее прогоняется весь массив данных, хранящийся на этом сервере. Полученные результаты могут либо сохраняться там же на сервере, либо отправляться для дальнейшей обработки.

Таким образом, на этапе обучения классификаторов о работе с Big Data пока речи не идет – не существует выборки такого объема, подготовленных для обучения систем, а на этапе классификации отдельные порции данных обрабатываются

независимо друг от друга. *Кластеризация* – ее задача состоит в разбиении множества информационных сущностей на группы, при этом члены одной группы более похожи друг на друга, чем члены из разных (классификация относит каждый объект к одной из заранее определенных групп). В качестве критерия схожести используется функция-расстояние, на вход которой поступают две сущности, а на выход – степень их схожести. Известно множество различных способов кластеризации (графовые, иерархические, итеративные, сети Кохонена).

Проблема кластеризации Big Data состоит в том, что имеющиеся алгоритмы предполагают возможность непосредственного обращения к любой информационной сущности в исходных данных (заранее невозможно предугадать, какие именно сущности понадобятся алгоритму). В свою очередь, исходные данные могут быть распределены по разным серверам, и при этом не гарантируется, что каждый кластер хранится строго на одном сервере. Если распределение данных по серверам делать прозрачным для алгоритма кластеризации, то это неизбежно приведет к копированию больших объемов с одного сервера на другой.

Решение проблемы может быть следующим. На каждом сервере запускается свой алгоритм, который оперирует только данными этого сервера, а на выходе дает параметры найденных кластеров и их веса, оцениваемые исходя из количества элементов внутри кластера. Затем полученная информация собирается на центральном сервере и производится метакластеризация – выделение групп близко расположенных кластеров с учетом их весов. Этот метод универсален, хорошо распараллеливается и может использовать любые другие алгоритмы кластеризации, однако он требует проведения серьезных научных исследований, тестирования на реальных данных и сравнения полученных результатов с другими «локальными» методами.

Таким образом, для анализа Big Data подавляющая часть методов кластеризации неприменима в чистом виде и необходимы дополнительные исследования.

Поиск закономерностей – суть метода заключается в нахождении правил, описывающих взаимозависимости между внутренними элементами данных. Классическим примером является анализ покупок в супермаркете и выявление правил вида «если человек покупает фотоаппарат, то обычно он покупает еще к нему аккумулятор и карту памяти». На вход задачи поиска закономерностей поступает неупорядоченное множество сущностей, для каждой из которых известен набор присутствующих информационных признаков; например, такими сущностями могут быть чеки

на покупки, а признаками – купленные товары. Задача поиска закономерностей сводится к выявлению правил вида «если присутствуют признаки A_1, A_2, \dots, A_n , то присутствуют и признаки B_1, B_2, \dots, B_m », при этом каждое правило характеризуется двумя параметрами: вероятностью срабатывания и поддержкой. Первый параметр показывает, как часто выполняется данное правило, а второй – как часто применимо данное правило, т.е. как часто встречается сочетание признаков A_1, A_2, \dots, A_n .

A/B тестирование — методика, в которой контрольная выборка поочередно сравнивается с другими. Тем самым удается выявить оптимальную комбинацию показателей для достижения, например, наилучшей ответной реакции потребителей на маркетинговое предложение. Большие данные позволяют провести огромное количество итераций и таким образом получить статистически достоверный результат.

Краудсорсинг – методика сбора данных из большого количества источников: категоризация и обогащение данных силами широкого, неопределённого круга лиц.

Смешение и интеграция данных – набор техник, позволяющих интегрировать разнородные данные из разнообразных источников для возможности глубинного анализа.

Машинное обучение («искусственный интеллект») – преследует цель создания алгоритмов самообучения на базе статистического анализа данных или машинного обучения для получения комплексных прогнозов.

Генетические алгоритмы – в этой методике возможные решения представляют в виде «хромосом», которые могут комбинироваться и мутировать. Как и в процессе естественной эволюции, выживает наиболее приспособленная особь.

Оптимизация – набор численных методов для редизайна сложных систем и процессов для улучшения одного или нескольких показателей. Помогает в принятии стратегических решений, например, состава выводимой на рынок продуктовой линейки, проведении инвестиционного анализа.

Визуализация аналитических данных – методы для представления информации в виде рисунков, графиков, схем и диаграмм с использованием интерактивных возможностей и анимации как для результатов, так и для использования в качестве исходных данных.

3.4. Алгоритмы кластеризации Big Data. При выполнении кластеризации основной проблемой является определение числа кластеров. Число методов разбиения множества на кластеры довольно велико. Все их можно подразделить на иерархические и неиерархические.

В неиерархических алгоритмах характер их работы и условие останковки необходимо заранее регламентировать часто довольно большим числом параметров, что иногда затруднительно, особенно на начальном этапе изучения материала. Но в таких алгоритмах достигается большая гибкость в варьировании кластеризации и обычно определяется число кластеров. С другой стороны, когда объекты характеризуются большим числом признаков (параметров), то приобретает большое значение задача группировки признаков.

В иерархических алгоритмах фактически отказываются от определения числа кластеров, строя полное дерево вложенных кластеров (дендрограмму). Их число определяется из предположений, не относящихся к работе алгоритмов, например, по динамике изменения порога расщепления (слияния) кластеров. Трудности таких алгоритмов хорошо изучены: выбор мер близости кластеров, проблема инверсий индексации в дендрограммах, негибкость иерархических классификаций, которая иногда весьма нежелательна. Тем не менее, представление кластеризации в виде дендрограммы позволяет получить наиболее полное представление о структуре кластеров.

3.4.1. Алгоритм *k*-средних (*k-means*).

Алгоритм *k-means* строит *k* кластеров, расположенных на возможно больших расстояниях друг от друга. Основной тип задач, которые решает алгоритм *k*-средних, – наличие предположений (гипотез) относительно числа кластеров, при этом они должны быть различны настолько, насколько это возможно. Общая идея алгоритма: заданное фиксированное число *k* кластеров наблюдения сопоставляется кластерам так, что средние в кластере максимально возможно отличаются друг от друга. Описание алгоритма [8]:

Этап 1. Первоначальное распределение объектов по кластерам. Выбирается число *k*, и на первом шаге эти точки считаются "центрами" кластеров. Каждому кластеру соответствует один центр. Выбор начальных центроидов может осуществляться следующим образом: выбор *k*-наблюдений для максимизации начального расстояния; случайный выбор *k*-наблюдений; выбор первых *k*-наблюдений. В результате каждый объект назначен определенному кластеру.

Этап 2. Вычисляются центры кластеров, которыми далее считаются по координатным средние кластеров. Объекты опять перераспределяются. Процесс вычисления центров и перераспределения объектов продолжается до тех пор, пока не выполнено одно из условий: кластерные центры стабилизировались, т.е. все наблюдения принадлежат кластеру, которому принадлежали до текущей итерации; число итераций равно максимуму.

Выбор числа кластеров является сложным вопросом. Если нет предположений относительно этого числа, рекомендуют создать 2 кластера, затем 3, 4, 5 и более, сравнивая полученные результаты.

Достоинства метода: простота использования; быстрота использования; понятность и прозрачность алгоритма.

Недостатки метода: алгоритм слишком чувствителен к выбросам, которые могут исказить среднее; медленная работа на больших базах данных; необходимо задавать количество кластеров.

3.4.2. Алгоритм HCM (Hard C – Means).

Метод *Hard C – Means* применяется, в основном, для кластеризации больших наборов числовых данных. Описание алгоритма [9]:

Шаг 1. Инициализация кластерных центров c_i ($i = 1, 2, \dots, c$). Это можно сделать, выбрав случайным образом c – векторов из входного набора.

Шаг 2. Вычисление рядовой матрицы M . Она состоит из элементов m_{ik} :

$$m_{ik} = \begin{cases} 1, & \|u_k - c_i\|^2 \leq \|u_k - c_j\|^2, \text{ для всех } i \neq j, \\ & i = 1, 2, \dots, c, k = 1, 2, \dots, K, \\ 0, & \text{остальное} \end{cases}$$

где K – количество элементов во входном наборе данных. Матрица M обладает следующими свойствами:

$$\sum_{i=1}^c m_{ij} = 1, \quad \sum_{j=1}^K m_{ij} = K.$$

Шаг 3. Расчет объектной функции:

$$J = \sum_{i=1}^c J_i = \sum_{i=1}^c \left(\sum_{k, u_k \in C_i} \|u_k - c_i\|^2 \right).$$

Шаг 4. Пересчет кластерных центров уравнением:

$$c_i = \frac{1}{|C_i|} \sum_{k, u_k \in C_i} u_k,$$

где $|C_i|$ – количество элементов в i -м кластере.

Шаг 5. Переход на шаг 2.

Достоинствами метода являются: легкость реализации, вычислительная простота, а недостатками – задание количества кластеров, отсутствие гарантии в нахождении оптимального решения.

3.4.3. Алгоритм нечеткой кластеризации Fuzzy C-means.

Метод нечеткой кластеризации *Fuzzy C-means* так же применяется, в основном, для кластеризации больших наборов числовых данных.

Описание алгоритма [10]:

Пусть нечеткие кластеры задаются матрицей разбиения:

$$F = [\mu_{ki}], \quad \mu_{ki} \in [0, 1], \quad k = \overline{1, M}; \quad i = \overline{1, c},$$

где μ_{ki} – степень принадлежности объекта k к кластеру i ; c – количество кластеров; M – количество элементов. При этом:

$$\left. \begin{aligned} \sum_{i=1}^c \mu_{ki} &= 1, \quad k = \overline{1, M}; \\ 0 < \sum_{k=1}^M \mu_{ki} &< M, \quad i = \overline{1, c}. \end{aligned} \right\} \quad (1)$$

Этап 1. Установить параметры алгоритма: c – количество кластеров; m – экспоненциальный вес, определяющий нечеткость, размазанность кластеров ($m \in [1, \infty)$); ε – параметр останова алгоритма.

Этап 2. Генерация случайным образом матрицы нечеткого разбиения с учетом условий (1).

Этап 3. Расчет центров кластеров:

$$V_i = \frac{\sum_{k=1}^M \mu_{ki}^m |X_k|}{\sum_{k=1}^M \mu_{ki}^m}, \quad i = \overline{1, c}.$$

Этап 4. Расчет расстояния между объектами X и центрами кластеров:

$$D_{ki} = \sqrt{\|X_k - V_i\|^2}, \quad k = \overline{1, M}; \quad i = \overline{1, c}.$$

Этап 5. Пересчет элементов матрицы разбиения с учетом следующих условий:

$$\text{если } D_k > 0: \mu_k = \frac{1}{\left(D_{jk}^2 * \sum_{j=1}^c \frac{1}{D_{jk}^2} \right)^{1/(m-1)}}, \quad j = \overline{1, c},$$

$$\text{если } D_k = 0: \mu_k = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases}, \quad j = \overline{1, c}.$$

Этап 6. Проверить условие $\|F - F^*\| < \varepsilon$, где F^* – матрица нечеткого разбиения на предыдущей итерации алгоритма. Если «Да», то переход к этапу 7, иначе к этапу 3.

Этап 7. Конец.

Достоинства: нечеткость при определении объекта в кластер позволяет определять объекты, которые находятся на границе, в кластеры. Недостатки: вычислительная сложность, задание количества кластеров, неопределенность с объектами, которые удалены от центров всех кластеров.

3.4.4. Алгоритм FRiS-Cluster.

Одной из основных проблем, возникающих при решении задачи классификации каких-либо объектов, является проблема выбора меры схожести. Чаще всего в этой роли выступает расстояние (евклидово, Чебышева, «городских кварталов»).

В основе алгоритма лежит FRiS-функция (Function of Rival Similarity – функция конкурентного сходства). FRiS – мера схожести двух объектов относительно некоторого третьего объекта. Она, в отличие от других существующих мер схожести, позволяет не просто оценивать понятия «далеко» или «близко», «похож» или «не похож», но и давать количественную оценку схожести. Такой подход позволяет учитывать большее число факторов при классификации. Исследования показали, что FRiS-функция хорошо имитирует человеческий механизм восприятия сходства и различия. Это позволяет использовать ее как базовый элемент для различных типов задач, включая задачи кластеризации документов.

Подробное описание алгоритма FRiS-Cluster можно найти в [11]. Алгоритм кластеризации можно описать следующим образом. Он последовательно ищет решение задачи кластеризации для значений $k = 1, \dots, K$, где K – заданное пользователем максимальное число кластеров. Затем из полученных решений выбирается лучшее. В описании алгоритма используются FRiS-функция:

$$F(a, S) = \frac{r_2(a, S) - r_1(a, S)}{r_2(a, S) + r_1(a, S)}$$

и редуцированная FRiS-функция:

$$F^*(a, S) = \frac{r_2^* - r_1(a, S)}{r_2^* + r_1(a, S)}.$$

Здесь: r_1 – минимальное расстояние до ближайшего столпа «своего» кластера, r_2 – минимальное расстояние до ближайшего столпа кластера-конкурента; r_2^* – расстояние до виртуального столпа (используется при $k=1$, когда настоящие конкуренты ещё не были найдены).

Шаг 1. При $k=1$ для каждого объекта a обучающей выборки находится значение средней редуцированной FRiS-функции $\bar{F}^*\{a\}$:

$$\bar{F}^*(S) = \frac{1}{M} \sum_{a \in A} F^*(a, S).$$

Шаг 2. Объект с максимальным значением средней редуцированной FRiS-функции принимается за первый столп – s_1 .

Шаг 3. Каждый объект a , за исключением s_1 , пробуете на роль второго столпа, столпом s_1 назначается объект, для которого значение функции $\bar{F}^*\{a, s_1\}$ оказывается максимальным.

Шаг 4. Все объекты исходной выборки распределяются между столпами s_1, s_2 , образуя кластеры A_1, A_2 , соответственно. Объект a относится к кластеру A_1 , если расстояние от a до s_1 меньше, чем для любого s_i .

Шаг 5. После добавления кластера A_2 , эталонный объект для кластера A_1 выбирается снова, но уже среди объектов кластера A_1 . Столпом s_{12} назначается объект, на котором достигает своего максимума функция $\bar{F}^*\{a, s_2\}$, описанная ниже. Аналогично, находится столп s_{22} для второго кластера. Полученная кластеризация представляет собой решение задачи для $k=2$:

$$\bar{F}(S) = \frac{1}{M} \sum_{a \in A} F(a, S).$$

Шаг 6. Далее для каждого нового столпа повторяются те же операции, что и при добавлении объекта два. Шаг 5 выполняется для всех кластеров текущей кластеризации.

Данный алгоритм имеет высокую трудоёмкость – $O(n^3)$, а также алгоритму требуется более ячеек n^2 памяти, что обусловлено необходимостью хранения матрицы парных расстояний.

3.4.5. Алгоритм FRiS-Stolp.

Другим применением FRiS-функции является один из алгоритмов отбора эталонных образцов

(столпов) для метрического классификатора, именуемый FRiS-Stolp. Выбор эталонов делается с помощью алгоритма FRiS-Stolp. Его идея состоит в том, что все объекты первого образа по очереди назначаются эталонами. Он нацелен на выбор минимального числа столпов, которые защищают не только самих себя, но обеспечивают заданную надежность защиты всех остальных объектов обучающей выборки. Первыми выбираются столпы, защищающие максимально возможное количество объектов с заданной надежностью. По этой причине при нормальных распределениях в первую очередь будут выбраны столпы, расположенные в точках математического ожидания [12].

Алгоритм FRiS-Stolp:

Шаг 1. Проверим вариант, при котором первый случайно выбранный объект a_i является единственным столпом образа S_1 , а все другие образы в качестве столпов имеют свои объекты. Для всех объектов $a_j \neq a_i$ первого образа находим расстояние r_{1j} до своего столпа a_i и расстояние r_{2j} до ближайшего объекта чужого образа. По этим расстояниям вычисляется значение FRiS-функции для каждого объекта a_j первого образа. Находим те m_i объектов первого образа, значение функций принадлежности F которых выше заданного порога F^* , например, $F^*=0$. По этим m_i объектам вычисляем суммарное значение FRiS-функции F_i , которое характеризует пригодность объекта a_i на роль столпа.

Шаг 2. Аналогичную процедуру повторяем, назначая в качестве столпа все M объектов первого образа по очереди.

Шаг 3. Находим объект a_i с максимальным значением F_i и объявляем его первым столпом A_{11} первого кластера S_{11} первого образа S_1 .

Шаг 4. Исключаем из первого образа m_i объектов, входящих в первый кластер.

Шаг 5. Для остальных объектов первого образа находим следующий столп повторением шагов 1–4.

Шаг 6. Процесс останавливается, если все объекты первого образа оказались включенными в свои кластеры.

Шаг 7. Восстанавливаем все объекты образа S_1 и для образа S_2 выполняем шаги 1–6.

Шаг 8. Повторяем шаги 1–7 для всех остальных образов.

На этом шаге заканчивается первый этап поиска столпов. Каждый столп A_i защищает подмножество объектов m_i своего кластера S_i .

4. Выводы

Эпоха Big Data уже наступила – объемы данных, генерируемых в науке, бизнесе, промышленности и управлении ИТ, растут экспоненциально. Однако существующие приложения обработки Big Data

не позволяют контролировать этапы ввода данных, собирать статистику и подбирать оптимальные структуры для хранения индексов, оптимизировать размещение данных на диске для обеспечения высокой скорости ввода/вывода, для выполнения аналитических запросов нет возможности произвести глубокий статистический анализ и выработать оптимальный план выполнения.

Важнейшее значение для масштабируемости и быстродействия приложений имеют характеристики сетей ЦОД — оптимизация качества обслуживания (QoS) требует активного обмена информацией между вычислительными узлами. Однако большинство нынешних ЦОД не способны обеспечить высокие скорости переноса данных, сопоставимые с показателями коммуникационных сетей высокопроизводительных компьютеров.

Что касается методов анализа для обработки Big Data, существующие на сегодня инструменты и наиболее распространенные методы анализа массивов данных пока не полностью удовлетворяют требованиям приложений обработки Big Data. В одном случае они не пригодны для обработки больших данных, в другом — затрудняется их применимость при построении автоматической классификации множества объектов в условиях отсутствия априорной информации о числе классов, в третьем — алгоритм имеет высокую трудоёмкость.

В данный момент можно прогнозировать высокоскоростную доставку данных из распределенных источников, оптимизацию переноса данных можно осуществить, например, с помощью развивающихся сейчас методов управления ресурсами с соблюдением гарантий качества обслуживания (QoS). В промышленной сфере можно прогнозировать аппаратные средства со специализированными датчиками для точного снятия показателей данных, а также развитие приложений, которые будут эти данные собирать, обрабатывать и структурировать, передавать в центры обработки, визуализировать для легкого восприятия, что, в свою очередь, облегчит принятие правильного решения.

В [13] разработана параллельная реализация алгоритма FRiS для кластеризации научных статей на основе технологии параллельных вычислений Message Passing Interface (MPI). В качестве меры близости при кластеризации принята мера конкурентного сходства. Для настройки весовых коэффициентов при вычислении меры сходства используется генетический алгоритм.

Литература: 1. Gantz John, Reinsel David. [http://www.emc.com/collateral/analyst-reports/idc-the-](http://www.emc.com/collateral/analyst-reports/idc-the-digital-universe-in-2020.pdf)

[digital-universe-in-2020.pdf](http://www.emc.com/collateral/analyst-reports/idc-the-digital-universe-in-2020.pdf). 2. Найдич А. www.compress.ru. 3. Селезнев К. Проблемы анализа Больших Данных // Открытые системы. СУБД №07, 2012 с.25-30. 4. Бабич В.К. и др. Основы металлургического производства. М.: Металлургия, 1988. 272 с. 5. Федин М.В. Перспективы использования систем обработки больших данных (bigdata) в металлургической промышленности // Economics. 2015, № 8(9). С. 52-54. 6. Проект Apache Hadoop. <https://hadoop.apache.org/>. 7. Артемов С. Big Data: новые возможности для растущего бизнеса.

<http://www.pcweek.ru/upload/iblock/d05/jet-big-data.pdf>.

8. Daniel Fasulo «An Analysis of Recent Work on Clustering Algorithms». Электронное издание. 9. Паклин Н. Алгоритмы кластеризации на службе Data Mining.

<http://www.basegroup.ru/clusterization/datamining.htm>.

10. Jan Jantzen «Neurofuzzy Modelling». Электронное издание. 11. Борисова И.А., Загоруйко Н.Г. Труды Всероссийской Конференции «Знания-Онтология-Теория». Новосибирск, 2007. Том II. С. 67-76. 12. Борисова И.А. и др. Труды Всероссийской конференции «Знания – Онтология – Теория», Новосибирск, 2007. Том I. С.37–44. 13. Мансурова М.Е. и др. Parallel computational technologies (PCT) 2016. <http://ceur-ws.org/Vol-1576/128.pdf>.

Transliterated bibliography:

1. Gantz John, Reinsel David. <http://www.emc.com/collateral/analyst-reports/idc-the-digital-universe-in-2020.pdf>.

2. Andrey Naydich. www.compress.ru.

3. Konstantin Seleznev. Problemy analiza Bolshih Danyih // Otkryitye sistemyi. SUBD #07, 2012 s.25-30.

4. Babich V.K. i dr. Osnovy metallurgicheskogo proizvodstva. M: Metallurgiya, 1988. 272 s.

5. Fedin M.V. Perspektivy ispol'zovaniya sistem obrabotki bol'shih danyih (bigdata) v metallurgicheskoy promyshlennosti // Economics. 2015, № 8(9). С. 52-54. 6. Project Apache Hadoop. <https://hadoop.apache.org/>.

7. Artemov S. Big Data: novyye vozmozhnosti dlya rastuschego biznesa.

<http://www.pcweek.ru/upload/iblock/d05/jet-big-data.pdf>.

8. Daniel Fasulo «An Analysis of Recent Work on Clustering Algorithms». Elektronnoe izdanie.

9. Paklin N. Algoritmyi klasterizatsii na sluzhbe Data Mining.

<http://www.basegroup.ru/clusterization/datamining.htm>.

10. Jan Jantzen «Neurofuzzy Modelling». Elektronnoe izdanie.

11. Borisova I.A., Zagoruyko N.G. Trudyi Vserossiyskoy Konferentsii «Znaniya-Ontologii-Teorii», Novosibirsk, 2007, Tom II, s. 67-76.

12. Borisova I.A. i dr. Trudyi Vserossiyskoy konferentsii «Znaniya-Ontologiya-Teoriya», Novosibirsk, 2007. Tom I. S.37–44.

13. Mansurova M.E. i dr. Parallel computational technologies (PCT) 2016. <http://ceur-ws.org/Vol-1576/128.pdf>.

Поступила в редколлегию 12.04.2017

Рецензент: д-р техн. наук, проф. Кривуля Г.Ф.

Магеррамов Закир Тулуевич, канд. техн. наук, доцент кафедры «Прикладная информатика» Азербайджанского Технического Университета. Научные интересы: численные методы, моделирование и оптимальное управление, информационные технологии, объектно-ориентированное программирование. Увлечения: научные книги, художественная литература (классика), мир животных. Адрес: Азербайджан, AZ1114, Баку, ул. И. Джумшудова, 1/7, кв. 110, тел. (99412)5689951, (050)3212595, e-mail: zakirmaharramov@rambler.ru

Абдуллаев Вугар Гаджимамудович, канд. техн. наук, доцент кафедры «Компьютерная инженерия технологии и программирование» Азербайджанской Государственной Нефтяной Академии (АГНА), Институт Кибернетики НАНА. Научные интересы: информационные технологии, веб-программирования, мобильные приложения. Увлечение: электронная коммерция, B2B, B2C проекты, научные книги, спорт. Адрес: Азербайджан, AZ1129, Баку, ул. М. Гади, 53, кв. 81, тел. (99412)5712428, (050)3325483, e-mail: abdulvugar@mail.com

Магеррамова Айнур Закировна, инженер-экономист, магистрант Aix-Marseille School of Economics (Франция). Научные интересы: численные методы в экономике, экономика развивающихся стран. Увлечения: конкурсы kaggle.com, йога. Адрес: Франция, 13100, Экс-ан-Прованс, ул. Маршал Леклерк, Л'Эстелан, кв. 205. E-mail: aynur.maharramova@etu.univ-amu.fr

Magerramov Zakir Tuluyevich, Cand. tech. Sci., Associate Professor of Applied Informatics at the Azerbaijan Technical University. Scientific interests: numerical methods, modeling and optimal control, information technologies, object-oriented programming. Hobbies: scientific books, artistic literature (classics), animal world. Address: Azerbaijan, AZ1114, Baku, I. Jumshudova, 1/7, apt. 110, tel. (99412) 5689951, (050) 3212595, e-mail: zakirmaharramov@rambler.ru

Abdullaev Vugar Gadzhimakhmudovich, Cand. tech. Sci., Associate Professor of Computer Engineering and Technology Programming at the Azerbaijan State Oil Academy (ASAN), Institute of Cybernetics of ANAS. Scientific interests: information technology, web programming, mobile application. Hobbies. e-commerce, B2B, B2C projects, science books, sports. Address: Azerbaijan, AZ1129, Baku, M. Gadi, 53, apt. 81, tel. (99412) 5712428, (050) 3325483, e-mail: abdulvugar@mail.com

Magerramova Ainur Zakirovna, engineer-economist, master student of the Aix-Marseille School of Economics (France). Scientific interests: numerical methods in economics, economics of developing countries. Hobbies: competitions kaggle.com, yoga. Address: France, 13100, Aix-en-Provence, Marshal Leklerk, L'Estelan, ap. 205. E-mail: aynur.maharramova@etu.univ-amu.fr