Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №6**

По дисциплине

«Компьютерное моделирование и Анализ данных»

за 7-й семестр

Выполнил:

студент 4 курса

группы ПО-5 (1)

Ворошин Д.Д.

Проверил:

Чичурин А. В.

Брест, 2022

**Визуальный анализ данных**

1. Выполнение визуального анализа данных.

Основной идеей визуального анализа данных является представление данных в некоторой визуальной форме, позволяющей человеку погрузиться в данные, работать с их визуальным представлением, понять их суть, сделать выводы и напрямую взаимодействовать с данными.

До недавнего времени визуальный анализ данных для отображения результатов на обычных мониторах использовал только двумерную или очень простую трехмерную графику. Однако прогресс в области аппаратных средств вывода изображений способствовал и совершенствованию средств визуального анализа данных. В настоящее время существует достаточно большое количество различных видов графических образов, позволяющих представлять результаты анализа в виде, удобном для понимания человеком.

За счет того, что пользователь напрямую работает с данными, представленными в виде визуальных образов, он может получить дополнительную информацию, которая поможет ему более четко сформулировать цели исследования.

Таким образом, визуальный анализ данных можно представить как процесс генерации гипотез. При этом сгенерированные гипотезы можно проверить или автоматическими средствами или средствами визуального анализа. Кроме того, прямое вовлечение пользователя в визуальный анализ имеет два основных преимущества перед автоматическими методами:

* визуальный анализ данных позволяет легко работать с неоднородными и зашумленными данными, в то время как не все автоматические методы могут работать с такими данными и давать удовлетворительные результаты;
* визуальный анализ данных интуитивно понятен и не требует сложных математических или статистических алгоритмов.

Следствием этих преимуществ является то, что визуальный анализ выполняется быстрее и в некоторых случаях дает лучший результат, чем автоматические методы анализа.

Визуальный анализ данных обычно выполняется в три этапа:

беглый анализ — позволяет идентифицировать интересные шаблоны и сфокусироваться на одном или нескольких из них;

увеличение и фильтрация — идентифицированные на предыдущем этапе шаблоны отфильтровываются и рассматриваются в большем масштабе;

детализация по необходимости — если пользователю нужно получить дополнительную информацию, он может визуализировать более детальные данные.

Так же как и при анализе данных, информация извлекается из некоторого источника, например, из базы данных или из файлов. Затем к ней могут быть применены [алгоритмы](https://intellect.icu/metodicheskoe-rukovodstvo-k-kursovoj-rabote-struktury-dannykh-i-algoritmy-4434#term-algoritmy) Data Mining для выявления скрытых закономерностей. Как результаты применения алгоритмов, так и исходные данные подвергаются обработке в ядре визуализации. Основной целью обработки является приведение многомерных данных к такому виду, который можно было бы представить на экране монитора.

В контексте визуализации данных Wolfram – достаточно мощный инструмент, который предполагает:

* **Графическое представление структурированных и неструктурированных данных.**Символьная архитектура и динамический интерфейс языка позволяют применять гибкие подходы к визуализации информации. В самом языке присутствует множество автоматизированных функций, настоящих «алгоритмов в алгоритме», которые позволяют загружать данные (как структурированные, так и неструктурированные) и тут же строить их графическое представление в динамическом режиме – от простых точечных диаграмм до графов и сетей.
* **Редактирование и контроль результатов визуализации в интерактивном режиме.** Wolfram предусматривает множество возможностей контроля над каждым этапом визуализации – от разработки и построения и до окончательного внешнего вида. Унифицированная символьная графическая архитектура языка позволяет программисту одновременно работать по двум направлениям – настраивать генерацию визуального представления в программном коде и интерактивно редактировать готовый результат. Разработчики назвали этот принцип «эстетической оптимизацией», когда пользователь может создавать визуализацию программно и затем настраивать ее на свой вкус. При этом имеется возможность «нарисовать» график или диаграмму с нуля и затем уже отрегулировать в коде программы.
* **Импорт и экспорт данных.**Wolfram поддерживает и автоматически обрабатывает данные практически любого известного формата; все они интегрированы в среду программирования в виде символьного обозначения. Для каждого формата данных связь между его представлениями внутри программы и вне ее может быть построена на любом уровне детализации – для этого в языке предусмотрен механизм работы с элементами данных.

1. Характеристики средств визуализации данных.

Существует достаточно большое количество средств визуализации данных. Рассмотрим более подробно три основные характеристики, которые позволяют определить наилучшее средство для визуализации данных:

* характер данных, которые нужно визуализировать с помощью данного средства;
* методы визуализации и образы, в виде которых могут быть представлены данные;
* возможности взаимодействия с визуальными образами и методами для лучшего анализа данных.

Наборы визуализируемых данных, как и в Data Mining, представляют собой матрицы, в которых ряды являются данными, а колонки — атрибутами данных. При этом данные могут характеризоваться одним или несколькими атрибутами. Кроме того, сами данные могут иметь более сложную структуру: иерархическую, текстовую, графическую и т. п. Таким образом, выделяют следующие виды данных, с которыми могут работать средства визуализации:

* одномерные данные — одномерные массивы, временные ряды;
* двумерные данные — точки двумерных графиков, географические координаты;
* многомерные данные — финансовые показатели, результаты экспериментов;
* тексты и гипертексты — газетные статьи, Web-документы;
* иерархические и связанные — структура подчиненности в организации, электронная переписка людей, гиперссылки документов;
* алгоритмы и программы — информационные потоки, отладочные операции.

Для визуализации перечисленных типов данных используются различные визуальные образы и методы их создания. Очевидно, что количество визуальных образов, которыми могут представляться данные, ограничиваются только человеческой фантазией. Основное требование к ним — это наглядность и удобство анализа данных, которые они представляют. Методы визуализации могут быть как самые простые (линейные графики, диаграммы, гистограммы и т. п.), так и более сложные, основанные на сложном математическом аппарате. Кроме того, при визуализации могут использоваться комбинации различных методов. Выделяют следующие типы методов визуализации:

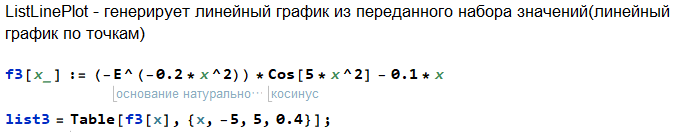
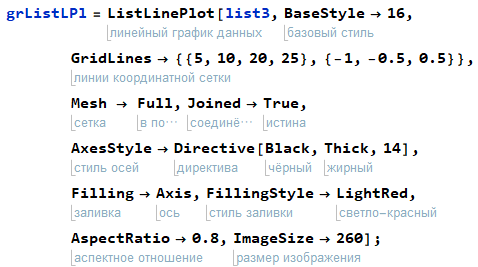
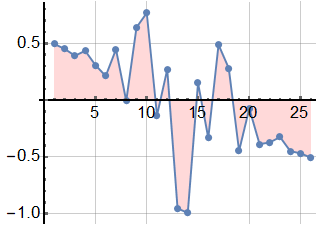
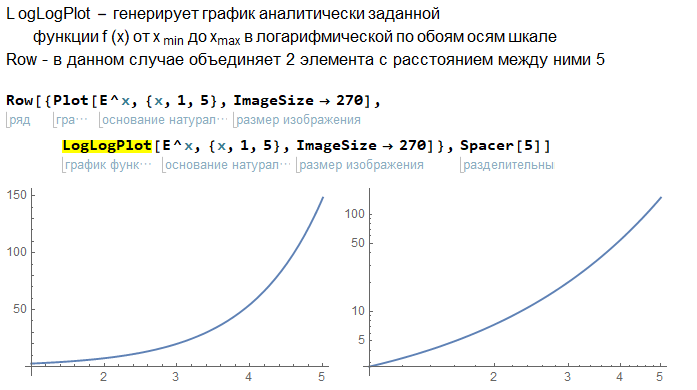
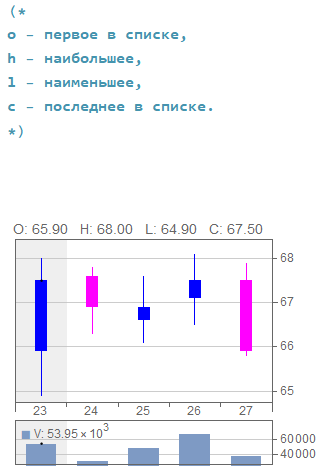
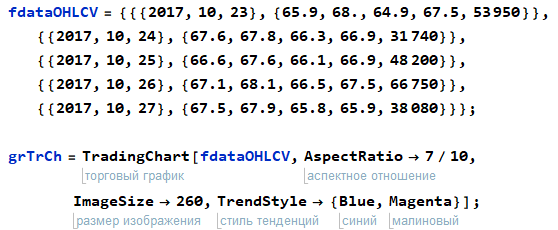
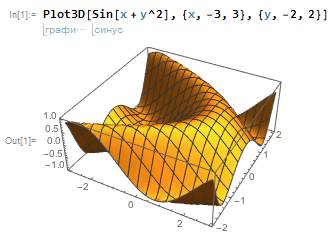
* стандартные 2D/3D-образы — гистограммы, линейные графики;
* геометрические преобразования — диаграмма разброса данных, параллельные координаты;
* отображение иконок — линейчатые фигуры (needle icons) и звезды (star icons);
* методы, ориентированные на пикселы — рекурсивные шаблоны, циклические сегменты;
* иерархические образы — древовидные карты и наложение измерений.

В результате применения методов визуализации будут построены визуальные образы, отражающие данные. Однако этого не всегда бывает достаточно для полного анализа. Пользователь должен иметь возможность работать с образами: видеть их с разных сторон, в разном масштабе и т. п.

Mathematica содержит полный набор функций для отображения структурированных и неструктурированных данных в 2d и 3d. Встроены функции для вывода графиков контурных и плотности, графиков из точек, линий и поверхностей, векторных графиков и графиков линий тока, гистограмм, двумерных и трехмерных секторных и столбчатых диаграмм, пузырьковых диаграмм, графиков для специализированных областей (таких как финансы и статистика, теория графов, управляющие системы и т.д.: график «японские свечи», Q-Q-нормальный вероятностный график (quantile plot), «ящик с усами» (box-and-whiskers plot), ЛАФЧХ (Bode plot) и многие другие).  
  
  
Графическая подсистема обеспечивает высококачественное статическое или динамическое представление функций, данных, диаграмм, изображений или аннотаций автоматически определяя баланс между скоростью вычислений и визуальным соответствием. Графические функции способны самостоятельно выбирать области печати и разбиение на регионы. Mathematica предоставляет сотни опций позволяющих контролировать каждый аспект отображения, включая размер рисунка, метки осей, отображение сетки, разбиение, заливку, 3d-освещение, угол наклона камеры и многое другое, позволяя создавать профессионально выглядящие графики. Более того, 3d-графики являются интерактивными, позволяющими изменять параметры камеры в реальном времени.

**Графическая визуализация в Wolfram**

Wolfram предоставляет огромное количество возможностей. Среди этих возможностей можно выделить:

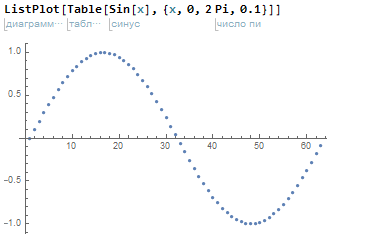
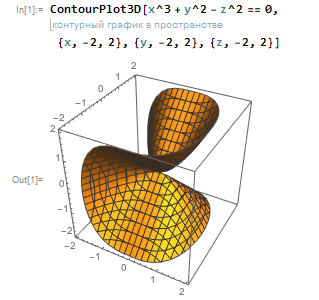
* построение кривых, иллюстрирующих поведение одной или нескольких функций одной переменной, заданных аналитическими выражениями или по точкам в декартовых, полярных координатах; функций, заданных в параметрической форме;   
  
* 
* 
* построение кривых, иллюстрирующих поведение функций с логарифмическими и полулогарифмическими масштабами;   
  
* построение столбиковых, секторных, финансовых диаграмм; представление данных, аппроксимация сплайнами, поддержка сплайнов, включая неоднородный рациональный В-сплайн;  
  
* разные способы отображения на плоскости трёхмерных поверхностей, задаваемых аналитически явными, неявными выражениями, в параметрической форме, а также наборами значений высот поверхности на равномерной сетке и рассеянном множестве точек;   
  
* разные способы оформления изображений трёхмерных поверхностей, в том числе с функциональной закраской, текстурным отображением изображений на трехмерных поверхностях;
* построение графиков векторных полей, линий тока, линий тока с фоном плотности функции; графиков вихря, градиента, Лапласиана векторного поля;
* построение проекций графиков поверхностей;
* построение пространственных фигур стереометрии, полиэдров;
* построение реалистических графиков рельефа;
* формирование трехмерных графических объектов, получаемых вращением кривых относительно некоторой оси;
* графы и сети полностью интегрированы в систему, включая построение графов, масштабируемые структуры данных, эстетическое оформление, широкие возможности для моделирования и анализа;
* построение графиков в основных картографических системах;
* возможности импорта рисунков и редактирования получаемых изображений инструментами встроенного редактора графики.

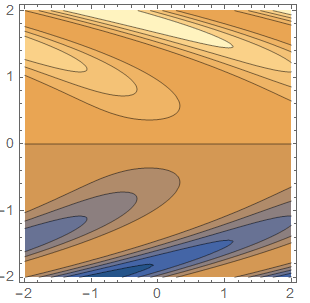
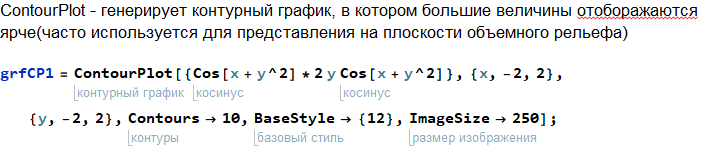
Существует достаточно много методов визуализации, но все они имеют как достоинства, так и недостатки. Основная идея комбинирования заключается в объединении различных методов визуализации для преодоления недостатков одного из них. Различные проекции рассеивания точек, например, могут быть скомбинированы с методами окрашивания и компоновки точек во всех проекциях. Такой подход может быть использован для любых методов визуализации. Окраска точек во всех методах визуализации дает возможность определить зависимости и корреляции в данных. Таким образом, комбинирование нескольких методов визуализации обеспечивает бóльшую информативность, чем в общем независимое использование методов. Типичными примерами визуальных образов, которые могут комбинироваться, являются: точки рассеивания, гистограммы, параллельные координаты, отображаемые пикселы и карты.

Любое средство визуализации может быть классифицировано по всем трем параметрам, т. е. по виду данных, с которым оно работает, по визуальным образам, которые оно может предоставлять, и по возможностям взаимодействия с этими визуальными образами. Очевидно, что одно средство визуализации может поддерживать разные виды данных, разные визуальные образы и разные способы взаимодействия с образами.

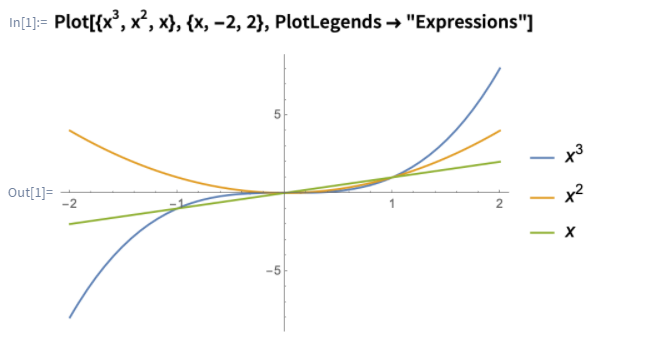
3) Методы визуализации.

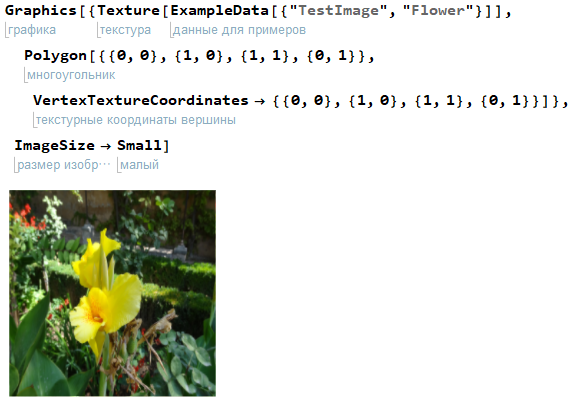
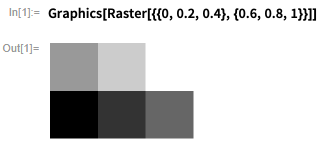
Одним из методов визуализации является метод геометрических преобразований. Основная идея методов геометрических преобразований — визуализировать преобразования и проекции данных в декартовом и в недекартовом геометрических пространствах. Wolfram позволяет выполнять следующие геометрические преобразования:

* точки и матрицы;  
  
* поверхностные и объемные графики, контуры;  
  



* параллельные координаты;



* текстуры и растры.  
    
  

4) Отображение иконок.

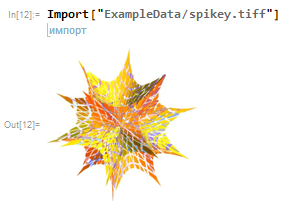
Подход, основанный на отображении иконок, предполагает каждому объекту данных ставить в соответствие некоторую иконку. При этом атрибуты объекта должны отображаться различными визуальными свойствами иконок. Иконки могут комбинироваться в матрицы или графики и, таким образом, предоставляют возможность анализировать все объекты в целом.

Wolfram позволяет очень гибко манипулировать самыми различными видами иконок. И позволяет как рисовать их из пикселей прямо в коде:

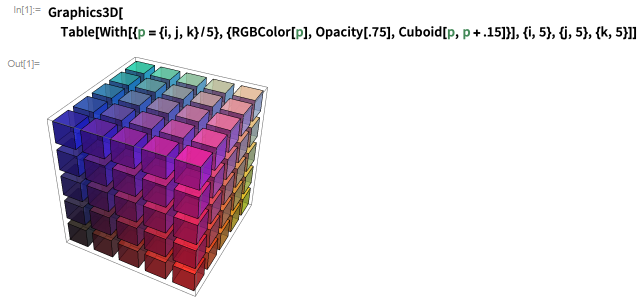




Так и импортировать файлы с изображениями:



А также использовать их при построении трёхмерных графиков:



5) Иерархические образы.

Иерархические образы используются для отображения иерархий и отношений в данных.

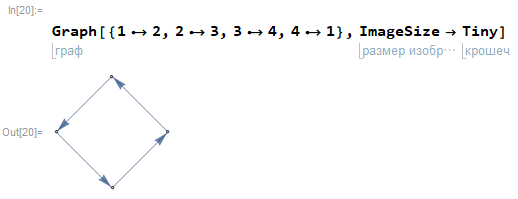
Wolfram позволяет работать с такими типами иерархических данных как:

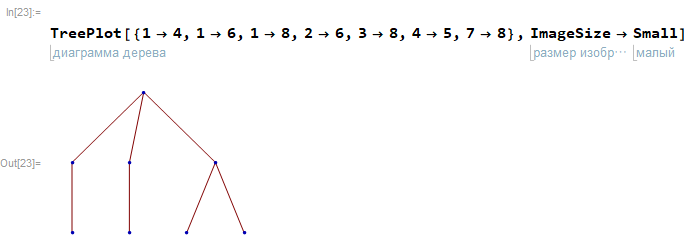
* Графы
* Деревья

Система*Mathematica* предоставляет современную функциональность для анализа и синтезирования графов и сетей. При работе с графами доступны следующие возможности:

* Поддержка ориентированных, неориентированных и взвешенных графов.
* Непосредственная поддержка случайных графовых распределений.
* Обширная коллекция операций на графах и операций по изменению графов.
* Поддержка теоретико-множественных операций на графах и операций, заданных булевыми выражениями.
* Выделение графовых элементов и подграфов с помощью языка шаблонов системы *Mathematica*.
* Всеобъемлющий набор предикатов для тестирования свойств графов.
* Эффективное тестирование изоморфности графов.
* Локальные и глобальные структурные свойства, включая компоненты накрытия и сопоставления.
* Более 15 метрик и мер центральности для квалифицирования графов и сетей.
* Эффективные функции нахождения кратчайшего пути, циклов и маршрутов.
* Мультипарадигмальный подход к графовому программированию с использованием матричной, оптимизационной или булевой подсистем.
* Общие алгоритмы поиска в глубину и в ширину, обладающие гибким программным интерфейсом.
* Поддержка произвольных свойств элементов графа.

Пример построения графа:



Деревья являются частными случаями графов, исходя из этого можно сделать вывод что для деревьев доступны такие же операции, как и для графов. Пример построения дерева: 

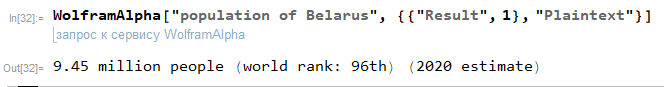
**Извлечение знаний из Web — Web Mining**

1. Проблемы анализа информации из Web.

Всемирная сеть сейчас содержит огромное количество информации, знаний. Пользователи на различных условиях могут просматривать всевозможные документы, аудио- и видеофайлы. Однако это многообразие данных скрывает в себе проблемы, которые могут возникнуть не только при анализе, но и при поиске необходимой информации в Интернет.

* Проблема поиска нужной информации связана с тем, что пользователь не всегда сразу может найти необходимые ему электронные ресурсы. Лишь небольшой процент ссылок среди предложенных поисковыми системами приводит к требуемым документам. Также труден поиск неиндексированной информации такими средствами.
* Проблема обнаружения новых знаний. Даже если найдено множество информации, для пользователя извлечение полезных знаний является довольно трудоемкой и непростой задачей. Сюда же можно и отнести сложности, связанные с осмыслением сведений, понятием тех идей, которые были вложены авторами.
* Проблема изучения потребителей связана с предоставлением пользователю информации, которая оказалась бы ему интересна. Это особенно актуально для электронных торговых порталов, которые могли бы "подсказывать" пользователю при выборе товара.

1. Этапы Web Mining.

Для решения перечисленных проблем используются различные технологии, напрямую или косвенно разрешающие их. К таким технологиям относятся: базы данных, информационный поиск, обработчики естественных языков и др. Технология Web Mining также направлена как на прямое, так и на косвенное решение перечисленных проблем  
  
Web Mining — технология, использующая методы Data Mining для исследования и извлечения информации из Web-документов и сервисов. Выделяют следующие этапы применения Web Mining:  
  
1. Поиск ресурсов — локализация неизвестных документов и сервисов в Web.  
2. Извлечение информации — автоматическое извлечение определенной информации из найденных Web-ресурсов.   
3. Обобщение — обнаружение общих шаблонов в отдельных и пересекающихся множествах сайтов.   
4. Анализ — интерпретация найденных шаблонов.   
  
  
  
  
  
  
  
  
  
**Пример:**  
Допустим нам нужно найти ценную информацию для того чтобы собрать какую-то статистику. Для этого можно создать простую задачу:  
Получить данные о том сколько людей было на земле за 2019-2020 год и сделать сравнение на сколько увеличилось население мира за год.  
  
Мы могли бы использовать информацию от Google или Википедию. но зачем когда есть WolframAlpha?  
  
**WolframAlpha** - это смесь алгоритмов и использование искусственного интеллекта для поиска нужной информации.  
  
Тоесть он сам найдёт на основе входных данных нужную мне информацию.   
Всё, что мне нужно будет сделать - это обработать её.  
  
  
Вот и всё.  
  
  
Поиск ресурсов предполагает поиск различных Web-источников (преимущественно текстовых) по ключевым словам. Данный этап разделяют на два класса: поиск документов и поиск сервисов.   
  
Большинство работ по поиску ресурсов сводится к автоматическому созданию поисковых индексов Web-документов. Для этих целей создавались роботы, индексирующие слова в документах и хранящие вычисленные индексы для дальнейшего их использования при обработке запросов пользователей. Наиболее популярными роботами считаются WebCrawler и AltaVista. Они способны сканировать миллионы документов и хранить индексы слов в этих документах. Существует много различных индексов, которые в настоящее время активно используются  
  
После того как ресурсы найдены, из них должна быть извлечена информация, подвергаемая анализу. Часто этот этап называют препроцессинг, т. к. он заключается в подготовке найденных ресурсов непосредственно к анализу. Такая подготовка заключается в преобразовании текстов, путем удаления стопслов, стеммингов, извлечением фраз и словосочетаний и т. п. Другими словами, результатом данного этапа должна быть информация, пригодная для анализа.   
  
На этапе обобщения к обработанной информации применяются методы Data Mining. На этом этапе важную роль играет человек, учитывая также тот факт, что на последнем этапе он должен будет интерпретировать полученные результаты.

1. Web Mining и другие интернет-технологии.

Web Mining, являясь инструментом для обработки и анализа Web-ресурсов, рассматривается в одном ряду с такими интернет-технологиями, как получение информации (Information Retrieval — IR) и извлечение информации (Information Extraction — IE). Однако, имея с ними много общего, Web Mining имеет также существенные отличия. Рассмотрим некоторые из них.

Технология IR заключается в получении документов из Web-среды, релевантных запросу пользователей (от англ. Relevant — относящийся к делу, что означает соответствие найденного ответа запросу, сделанному пользователем поисковой системы). При этом очень часто полученные документы включают в себя как релевантные, так и нерелевантные документы. Как уже упоминалось, для решения этой задачи строятся поисковые индексы. Для их построения используются различные методы, включающие в себя моделирование, классификацию и кластеризацию документов, фильтрацию и др. При этом для классификации и кластеризации документов могут быть использованы методы Data Mining (точнее Text Mining). С этой точки зрения можно говорить, что Web Mining является частью технологии IR.

Целью IE является извлечение необходимых фактов из Web-документов. Основное отличие этой технологии от IR заключается в том, что она работает с самим документом и ищет в нем релевантную информацию, в то время как IR работает с множеством документов, извлекая из него релевантные документы.

IE основное внимание уделяет структуре текстового документа, пытаясь извлечь из него ключевые понятия. Таким образом, IE и Web Mining может находиться в разных отношениях друг к другу. С одной стороны, для исследования структуры и извлечения основных понятий из текста могут быть использованы методы Text Mining. В этом случае Web Mining является частью IE. С другой стороны, структуризация документов методами IE позволяет сохранять его в реляционной базе данных, что, в свою очередь, позволяет применить к нему методы Data Mining. Таким образом, IE может рассматриваться как технология препроцессинга на одном из этапов Web Mining.

Как следует из анализа, различные методы и технологии могут использоваться совместно, взаимно улучшая друг друга.

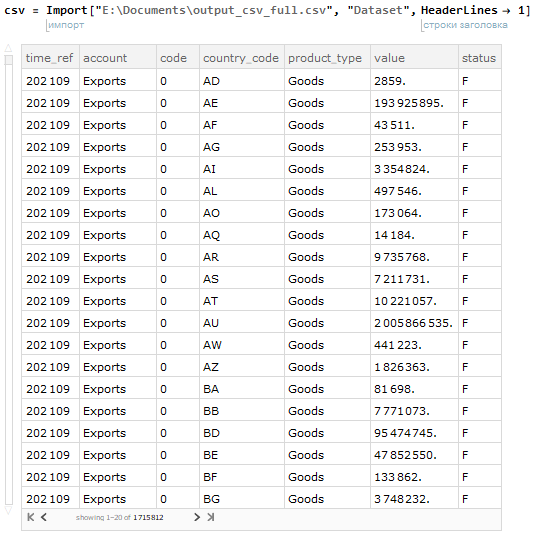
1. Извлечение Web-контента в процессе информационного поиска.

Методы извлечения Web-контента в процессе информационного поиска во многом зависят от типа анализируемых документов. Различают два основных типа: неструктурированные и почти структурированные. К неструктурированному типу относятся все текстовые документы, не имеющие определенной структуры. К почти структурированным относятся документы, имеющие структуру в целом, но позволяющую вхождение в структурный элемент неструктурированного текста. К таким документам относятся HTML, XML и др. Большинство методов анализа неструктурированного текста использует представление текстового документа в виде множества или вектора слов. Данный подход также широко применяется в методах Text Mining. При этом в такие представления помещаются отдельные слова без учета их расположения, связи с другими словами, контекста и других лингвистических особенностей. Каждому слову во множестве ставится в соответствие некоторое свойство. Данное свойство может иметь или логический тип, отражающий наличие или отсутствие слова в тексте, или числовое значение, отражающее частоту появления слова в тексте. Последующая обработка может быть связана с удалением пунктуации, нечастых слов, стоп-слов и др. Уменьшение числа свойств возможно за счет применения различных методов выбора свойств, основанных на расчете следующих метрик:   
  
- информационного прироста (information gain);   
- полного количества информации (mutual information);   
- перекрестной энтропии (cross entropy);   
- вероятности успешного исхода (odds-ration).  
  
Кроме большого размера модели, векторное представление документов имеет еще один существенный недостаток: оно не обрабатывает синонимы — документы считаются семантически далекими друг от друга, если в них нет одинаковых слов. Данный недостаток устраняется методом скрытой семантической индексации (Latent Semantic Indexing — LSI). Согласно этому методу, пространство термов сингулярным разложением (отбрасываются наименее значимые сингулярные значения) приводится к пространству ортогональных факторов (некоррелируемых "индексных термов"). Поскольку результирующие факторы играют роль "приведенных" термов, декомпозиция "сближает" документы из одинаковых предметных областей. В результате документы одной тематики, но которые не используют одинаковые термины, размещаются в одном разделе, и их стемминг сокращает слова с общими морфологическими корнями. Например, слова "информирование", "информация", "информатор" и "информировано" приводятся к их общему корню "информ" — и только это слово используется как свойство документа, вместо четырех форм.

Необходимо обратить внимание, что эффективность вариантов препроцессинга, направленных на сокращение размера набора свойств, может быть различна для разных областей.

Кроме представления документа в виде вектора слов, возможны и другие представления:   
  
- использующие информацию о позиции слова в документе;   
- использующие n-граммное представление (последовательности слов длины вплоть до n) (например, "морфологический корень" — 3-грамма), использующие целые фразы (например, "быстрая лиса исчезла из вида");  
- использующие понятие документа категорий;  
- использующие термины (например, "норма годового процента" или "Уолл-стрит");  
- использующие гипернимы (hypernym — слово, являющееся более общим, абстрактным по отношению к данному) (лингвистический термин отношения "это есть" — "собака есть животное", поэтому "животное" — это hypernym "собаки");  
- использующие адресные объекты (например, имена людей, даты, почтовые адреса, расположения, организации или URL).

Реляционное представление является по сути первым представлением, учитывающим порядок слов. Данное представление более выразительно, чем позиционная логика. Например, во множественном представлении слову соответствует его частота. В представлении в виде отношений используется взаимосвязь между различными словами и их позициями, например "слово X левее слова Y в том же предложении". Несмотря на то, что широко используются различные виды представлений естественных языков, в настоящий момент нет исследований, которые наглядно показали бы преимущества тех или иных видов представлений для разных типов задач текстовой классификации. Скотт и Матвин сравнивают различные представления (множество слов, представление, основанное на фразах, гипернимы и др.), но они не нашли никаких их существенных достоинств или недостатков по отношению друг к другу.   
  
Таблица:   
  
В силу схожести задачи извлечения Web-контента с задачами Text Mining для ее решения используются методы, подробно описанные в таблице ниже. В этой таблице представлены методы, применяемые для решения задачи извлечения Webконтента. В ней для каждого метода приведены модели представления документов и прикладные задачи, в которых они могут быть использованы. Методы извлечения Web-контента находят применение в разных задачах: текстовой классификации и кластеризации; определения событий и трекинга; поиска извлекаемых шаблонов или правил; поиска шаблонов в текстовых документах. Отдельно стоит остановиться на задаче определения событий и трекинге. Данная задача является частной задачей более широкого направления автоматизированной обработки новостных данных — Topic Detection and Tracking (TDT) . В TDT выделяют следующие направления исследований:  
  
- разбиение потока на сюжеты;  
- идентификация новых событий;  
- определение связей между новостными историями;  
- отслеживание интересующей пользователя информации.   
  
5) Слабоструктурированные документы.   
  
Извлечение Web-контента из слабоструктурированных документов использует более развитые средства представления текста. Это в первую очередь связано с тем, что в документах уже выделены некоторые структурные элементы. Практически все методы в этой области для представления документа используют HTML-структуры внутри документов. Некоторые методы используют также для представления гиперссылки между документами.



Как и в случае с неструктурированными документами, к полученным представлениям применяются общие методы Data Mining.

Область применения методов довольно широка:

- гипертекстовая классификация;

- классификации и кластеризации;

- изучение отношений между Web-документами;

- извлечение шаблонов или правила;

- поиск шаблонов и слабоструктурированных данных.

6) Извлечение Web-контента для формирования баз данных.

Задача извлечения Web-контента для его размещения в базе данных относится к проблеме управления информацией и обработки запросов к ней. Существуют три класса задач, относящихся к этой проблеме:

моделирование и формирование запросов к Web;

извлечение информации и интеграция;

создание и реструктуризация Web-сайта.

Хотя первые две задачи относятся к категории извлечения Web-контента, не все методы, применяемые при их решении, относятся к этой категории. Это связано с отсутствием машинного обучения или использованием методов Data Mining в процессе их решения. Обычно методы извлечения Web-контента пытаются выявить структуру Web-документа или преобразовать его для сохранения в базе данных таким образом, чтобы улучшить информационное управление и cделать возможным запрос к нему.

С точки зрения размещения Web-контента в базе данных целью, большей частью, является построение модели данных и объединение их таким образом, чтобы поиск мог выполняться не только по ключевым словам, но и по запросам, более приближенным к естественному языку. Этого можно достичь построением схемы Web-документов, формированием хранилища, базы знаний или виртуальной базы данных. Исследования в этой области большей частью имеют дело со слабоструктурированными данными. Слабоструктурированные данные из представления базы данных часто ссылаются на данные, которые имеют некоторую структуру, но не жесткую схему.

Из таблицы можно видеть, что методы извлечения Web-контента для целей базы данных используют представления, которые отличаются от представлений, используемых для целей информационного поиска. Данные методы в основном используют представления в виде модели объектного обмена (Object Exchange Model — OEM) .

Информация в OEM-структуре представляется в виде графа с именованными ребрами, а узлы соответствуют объектам. Для определенности будем называть их node-объектами или n-объектами. В простейшем варианте они могут быть атомарными или контейнерами. Атомарные n-объекты имеют только входящие связи и значение определенного типа, но не имеют зависимых объектов. У контейнеров нет значений, но есть зависимые объекты (исходящие связи). Каждый n-объект имеет уникальный идентификатор. На связи между ними, в общем случае, не накладывается никаких ограничений.

В большинстве своем рассмотренные методы применяются в задачах, связанных с выявлением, исследованием или формированием схем DataGuides .

DataGuide — сжатый вид схемы слабоструктурированных данных. Для практического применения и из-за сложности вычислений DataGuide часто аппроксимируется. Некоторые приложения не решают задачу поиска глобальной схемы, они концентрируются на задачах поиска подсхем в слабоструктурированных данных.

(Таблица) Методы извлечения Web-контента из слабоструктурированных документов в сохранения в базе данных

|  |  |  |  |
| --- | --- | --- | --- |
| Методы | Авторы | Представление | Применение |
|  |  | документа |  |
|  |  |  |  |
| Частные алгоритмы | Р. Голдман | OEM | Поиск DataGuide |
|  | (R. Goldman) |  | в слабоструктурированных |
|  | и Дж. Видом |  | данных |
|  | (J. Widom) |  |  |
|  |  |  |  |
|  | Ш. Грумбах | Строки | Поиск схемы |
|  | (S. Grumbach) | и отношения | в слабоструктурированных |
|  | и Г. Мекка |  | данных |
|  | (G. Mecca) |  |  |
|  |  |  |  |
|  | С. Нестеров | OEM | Поиск типов иерархии |
|  | (S. Nestorov) |  | в слабоструктурированных |
|  | и др. |  | данных |
|  |  |  |  |
| Модифицированные | Х. Тоивонен | OEM | Поиск используемых |
| алгоритмы поиска | (H. Toivonen) |  | подструктур |
| ассоциативных |  |  | в слабоструктурированных |
| правил |  |  | данных |
|  |  |  |  |
|  | Х. Лиу (H. Liu) | OEM | Поиск шаблонов |
|  | и К. Ванг |  | в слабоструктурированных |
|  | (K. Wang) |  | данных |
|  |  |  |  |
| Атрибутно- | О. Зайн (O. Zaiane) | Отношения | Мультиуровневая база |
| ориентированное | и Дж. Хан (J. Han) |  | данных |
| утверждение |  |  |  |
|  |  |  |  |

Другое применение методов данной категории Web Mining — это формирование многослойной базы данных (MLDB), в которой каждый уровень создается обобщением низших уровней и использует специальный язык запросов для Web Mining, чтобы извлекать некоторые знания из MLDB.

Из-за различий представления Web-документов, используемых в извлечении Web-контента для их сохранения в БД, большинство из используемых методов отличаются от классических методов Data Mining, которые действуют на плоских данных и требуют дополнительной модификации. Исключение составляют методы ILP, которые могут работать с отношениями и графическим представлением данных. Для того чтобы действовать на реляционных или графических данных, используют вероятностные алгоритмы для обнаружения схем и для конструкции MLDB, применяют измененную версию алгоритмов построения ассоциативных правил и модифицированную версию порядковой логики правил ассоциации .

Необходимо отметить, что существует также большая область, связанная с анализом мультимедийной информации, которая в большом объеме присутствует в Web-документах. Методы, позволяющие извлекать полезные знания из мультимедийной информации, объединяются в технологию Multimedia Mining и требуют отдельного рассмотрения

1. Исследование использования Web-ресурсов.  
     
   Процесс исследования использования Web-ресурсов обычно включает в себя только три фазы:  
     
   1. Препроцессиг  
   2. Извлечение шаблонов  
   3. Анализ шаблонов.  
     
   В отличие от предыдущих двух задач Web Mining, в этой задаче исходными данными являются не сами страницы и их содержимое, а информация, записываемая в результате взаимодействия пользователей с Web. Выделяют следующие типы данных, применяемые в задаче исследования использования Web-ресурсов:  
   использование — данные, которые описывают использование страниц, такие как IP-адресса, ссылки на страницы, а также дату и время доступа к ним;  
   пользовательские профили — данные, которые обеспечивают демографическую информацию (пол, возраст, социальное положение и др.) о пользователе, а также регистрационную информацию.  
   Данные об использовании Web собираются в различных источниках, которые можно разделить на следующие основные группы: серверы, клиенты, прокси.

На стороне сервера информация для анализа может извлекаться из логов, трафика сервера, куки-файлов, запросов пользователей и др.

Логи Web-серверов являются важным источником информации, т. к. они в явном виде содержат описание действий посетителей сайтов. Однако до сервера доходит не вся информация о действиях пользователей, так просмотр пользователем страниц, сохраненных у него в кэше, никак не отражается на сервере. Кроме того, информация, заключенная в HTTP Post-запрос, не будет сохраняться в логе сервера.  
  
Для извлечения информации из входящего в сервер трафика может использоваться технология анализатора пакетов (packet sniffer). Данная технология позволяет извлекать информацию напрямую из TCP/IP-пакетов, поступающих на Web-сервер.

Куки-файлы автоматически генерируются для каждого пользовательского браузера, чтобы поддержать сессионное взаимодействие. Они хранят информацию, идентифицирующую пользовательскую сессию.

Пользовательские запросы содержат информацию о потребностях пользователей, искомых документах и др.

На стороне клиента потенциально может находиться достаточно много информации о поведении пользователей, начиная от просматриваемых страниц и заканчивая щелчками мыши. Однако основная трудность заключается в извлечении этой информации, т. к. пользователей сайтов бесконечно много и заранее невозможно предсказать, какие пользователи будут обращаться к сайтам.

В литературе предлагаются подходы, основанные на использовании агентов (реализованных как Java-апплеты или Java-скрипты) и на изменении кода Web-браузеров. Подход, использующий агентов, предполагает его передачу на сторону клиента, где он будет записывать и отправлять на сервер информацию о действиях пользователя.

Такой подход имеет ряд недостатков:

- Java-апплеты и Java-скрипты имеют ограниченные возможности, выполняясь в рамках браузера (так называемой песочницы);

- не мониторятся действия пользователей до момента начала работы агента (время загрузки и время инициализации агента);

- отслеживаться может только взаимодействие одного пользователя (которому загружен агент) с одним сайтом (с которого загружен агент).

Подход, основанный на модификации существующих браузеров с целью встроить в них средства мониторинга действий пользователя, позволяет решить перечисленные проблемы. Однако он сталкивается с трудностями, связанными с тем, что не все браузеры являются Open Source (т. е. предоставляют исходный код), а следовательно, могут быть модифицированы. К браузерам с открытым кодом относятся такие популярные, как Mozaic или Mozilla. Ввиду этого появляется проблема, связанная с убеждением пользователей использовать именно такие модифицированные браузеры.

Прокси-серверы являются промежуточным уровнем между клиентом и сервером и используются для кэширования часто запрашиваемых пользователем страниц. Они могут хранить действительные запросы от нескольких клиентов к различным сайтам, которые не дошли до серверов. Такая информация может рассматриваться как поведение группы анонимных пользователей.

Ни один из видов источников не хранит в себе полной информации, а следовательно, не может рассматриваться как единственный поставщик данных для анализа. Для повышения качества анализа поведения пользователей в нем должна участвовать информация от всех видов источников данных.

Вся информация, поставляемая от перечисленных ранее источников, может быть описана в следующих терминах, определенных группой W3C Web Characterization Activity (WCA).

**Пользователь** (user) — индивидуум, получающий доступ к файлу одного или нескольких Web-серверов через браузер. Сложность данного понятия связана с идентификацией уникального пользователя, т. к. он может получать доступ с разных клиентских машин, а также от его имени могут работать различные агенты.

**Просмотр страницы** (page view) — включает в себя одновременное отображение нескольких файлов в браузере пользователя. Данное понятие ассоциирует одного пользователя и несколько файлов (объединенных в одной Web-странице, в том числе и фреймовой структурой), открываемых одним щелчком.

**Потоки кликов** (click streams) — последовательность открытия Web-стра- ниц (вызываемых кликами). Не всегда можно восстановить полную последовательность, основываясь только на информации со стороны сервера, т. к. запросы, обрабатываемые клиентом или прокси-сервером, "не известны".

**Пользовательская сессия** — последовательность просматриваемых страниц (поток кликов) одним пользователем на разных Web-сайтах.

**Сессии сервера** (server sessions) — набор просматриваемых страниц в рамках одной пользовательской сессии с одного Web-сервера. Для анализа обычно доступна только серверная сессия, ограниченная одним сервером.

**Эпизоды** (episodes) — любой семантически значимый поднабор пользовательской или серверной сессии.

1. Этап препроцессинга.

Для решения задачи исследования использования Web на этапе препроцессинга в массиве анализируемых данных должны быть выделены перечисленные сущности. Это значительно усложняет неполнота данных, получаемых с одного источника. Например, для идентификации пользователя недостаточно только IP-адреса, т. к. возможны следующие ситуации.

- Один IP-адрес — несколько серверных сессий. Интернет-провайдеры обычно имеют пул прокси-серверов, доступных пользователям. В одно и то же время пользователь может просматривать страницы одного сайта с различных прокси-серверов.

- Несколько IP-адресов — одна серверная сессия. Некоторые интернетпровайдеры или инструменты безопасности случайным образом каждый пользовательский запрос ассоциируют с разными IP-адресами. В результате одна серверная сессия ассоциируется с несколькими IP-адресами.

- Несколько IP-адресов — один пользователь. Пользователь может получать доступ к Web-ресурсам с разных машин, а следовательно, будет ассоциирован с разными IP-адресами. Это усложняет отслеживание повторяющихся запросов от одного пользователя.

- Множество агентов — один пользователь. Пользователь, работающий с более чем одним браузером или пользующийся агентами, даже с одной машины будет распознаваться как несколько пользователей.

При условии если удалось идентифицировать пользователя (с помощью кукифайлов, логина или анализа его IP/агента/пути), необходимо поток кликов от каждого пользователя разделить на сессии. Эта задача усложняется тем, что тяжело определить, когда пользователь покинул сайт, т. к. обычно при анализе запросы данного пользователя к другим серверам не доступны. Кроме того, часто сессия прерывается по тайм-ауту, если пользователь не возобновляет

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | Т а б л и ц а | | Пример серверного лога | | |
|  |  |  |  |  |  |  |  |  |  |
| № | IP-адрес | Время | Метод | URL | Протокол | Статус |  | Ссылка | Браузер |
|  |  |  |  |  |  |  |  |  |  |
| 1 | 123.456.78.9 | 25/Apr/1998:03:04:41 -0500 | GET | A.html | HTTP/1.0 | 3290 |  |  | Mozilla/3.04 (Win95, I) |
|  |  |  |  |  |  |  |  |  |  |
| 2 | 123.456.78.9 | 25/Apr/1998:03:05:34 -0500 | GET | B.html | HTTP/1.0 | 2050 |  | A.html | Mozilla/3.04 (Win95, I) |
|  |  |  |  |  |  |  |  |  |  |
| 3 | 123.456.78.9 | 25/Apr/1998:03:05:39 -0500 | GET | L.html | HTTP/1.0 | 4130 |  |  | Mozilla/3.04 (Win95, I) |
|  |  |  |  |  |  |  |  |  |  |
| 4 | 123.456.78.9 | 25/Apr/1998:03:06:02 -0500 | GET | F.html | HTTP/1.0 | 5096 |  | B.html | Mozilla/3.04 (Win95, I) |
|  |  |  |  |  |  |  |  |  |  |
| 5 | 123.456.78.9 | 25/Apr/1998:03:06:58 -0500 | GET | A.html | HTTP/1.0 | 3290 |  |  | Mozilla/3.01 |
|  |  |  |  |  |  |  |  |  | (X11, I, IRIX6.2, IP22) |
|  |  |  |  |  |  |  |  |  |  |
| 6 | 123.456.78.9 | 25/Apr/1998:03:07:42 -0500 | GET | B.html | HTTP/1.0 | 2050 |  | A.html | Mozilla/3.01 |
|  |  |  |  |  |  |  |  |  | (X11, I, IRIX6.2, IP22) |
|  |  |  |  |  |  |  |  |  |  |
| 7 | 123.456.78.9 | 25/Apr/1998:03:07:55 -0500 | GET | R.html | HTTP/1.0 | 8140 |  | L.html | Mozilla/3.04 (Win95, I) |
|  |  |  |  |  |  |  |  |  |  |
| 8 | 123.456.78.9 | 25/Apr/1998:03:09:50 -0500 | GET | C.html | HTTP/1.0 | 1820 |  | A.html | Mozilla/3.01 |
|  |  |  |  |  |  |  |  |  | (X11, I, IRIX6.2, IP22) |
|  |  |  |  |  |  |  |  |  |  |
| 9 | 123.456.78.9 | 25/Apr/1998:03:10:02 -0500 | GET | O.html | HTTP/1.0 | 2270 |  | F.html | Mozilla/3.04 (Win95, I) |
|  |  |  |  |  |  |  |  |  |  |
| 10 | 123.456.78.9 | 25/Apr/1998:03:10:45 -0500 | GET | J.html | HTTP/1.0 | 9430 |  | C.html | Mozilla/3.01 |
|  |  |  |  |  |  |  |  |  | (X11, I, IRIX6.2, IP22) |
|  |  |  |  |  |  |  |  |  |  |
| 11 | 123.456.78.9 | 25/Apr/1998:03:12:23 -0500 | GET | G.html | HTTP/1.0 | 7220 |  | B.html | Mozilla/3.04 (Win95, I) |
|  |  |  |  |  |  |  |  |  |  |
| 12 | 209.456.78.2 | 25/Apr/1998:05:05:22 -0500 | GET | A.html | HTTP/1.0 | 3290 |  |  | Mozilla/3.04 (Win95, I) |
|  |  |  |  |  |  |  |  |  |  |
| 13 | 209.456.78.3 | 25/Apr/1998:05:06:03 -0500 | GET | D.html | HTTP/1.0 | 1680 |  | A.html | Mozilla/3.04 (Win95, I) |
|  |  |  |  |  |  |  |  |  |  |

взаимодействие с сервером через некоторый промежуток времени. Если ID сессии встраивается в каждый URI, то определение сессии устанавливается контентным сервером. В этом случае запрашиваемый контент хранится в поле запроса и сохраняется в логе на сервере. Однако в связи с тем, что для каждой сессии сервер может иметь переменные состояний, в которых хранит некоторую информацию о сессии, то не вся она доступна из URI. Также использование кэша и связанное с этим недохождение запросов до сервера усложняет выделение сессий. Эта проблема может быть решена только мониторингом запросов от клиентской стороны к кэшу. Ссылочные поля запросов могут быть использованы, чтобы определить, когда были просмотрены страницы из кэша.

В таблице приведен пример серверного лога (первая колонка добавлена для удобства), на котором можно увидеть описанные ранее проблемы. IP-адрес 123.456.78.9 относится к трем серверным сессиям. Адреса 209.456.78.2 и 209.45.78.3 относятся к одной — четвертой сессии. Выделение 3-х сессий для одного адреса (с 1-й по 11-ю строку таблицы) возможно на основании совместно используемой информации о ссылке, по которой произошел переход с предыдущей страницы, и агенте. В результате запросы разбиваются на следующие сессии: A-B-F-O-G, L-R и A-B-C-J. Если сделать путь переходов непрерывным и добавить возвраты на предыдущие страницы, то в первую сессию добавятся еще две страницы A-B-F-O-F-B-G и одна страница добавится к третьей сессии A-B-A-C-J.

Вданном примере без использования информации о куки-файлах, встроенной в запрос ID сессии, или информации со стороны клиента не удастся определить, что строки 12 и 13 принадлежат одной сессии, т. к. они различаются IP-адресами.

На этапе препроцессинга кроме выделения основных элементов, необходимых для дальнейшего анализа, часто выполняется фильтрация по контексту. Такая задача может возникнуть, например, если аналитика интересует использование Web-ресурсов по определенной тематике. Для такой фильтрации применяют методы анализа Web-контента.

1. Этап извлечения шаблонов.   
     
   Для извлечения шаблонов из информации об использовании Web-ресурсов применяются различные методы как классической статистики, так и относящиеся к области Data Mining.  
     
   Методы статистики часто используются для анализа посещения сайтов и трафиков. Так, анализ сессионных файлов позволяет выполнить различные виды дескриптивного статистического анализа (вычислить частоту, матема  
     
     
   тическое ожидание, дисперсию и т. п.) для просмотров страниц, времени просмотра и длины навигационного пути. Многие инструменты анализа трафика позволяют получить такие характеристики, как наиболее часто посещаемые страницы, среднее время посещения страниц или средняя длина пути перемещения по страницам. Подобные отчеты могут содержать низкоуровневые ошибки, связанные с невозможностью определить неавторизированные точки входа или недействительные URI.  
     
   Такой вид получаемых знаний может быть весьма полезным для улучшения производительности систем, повышения безопасности систем, решения задач модификации сайтов и обеспечения поддержки для решения маркетинговых задач.  
     
   Методы генерации ассоциативных правил могут быть использованы для выявления наиболее часто совместно запрашиваемых страниц, объединенных одной серверной сессией. Эти страницы могут быть связаны не напрямую друг с другом (т. е. не иметь прямых ссылок друг на друга). Например, алгоритмом Apriori может быть обнаружена корреляция между пользователем, посетившим страницу с электронной продукцией, и пользователем, просматривающим страницу со спортивным оборудованием.  
     
   Наличие или отсутствие таких правил в области бизнеса и маркетинга может помочь Web-дизайнерам перестроить Web-сайт. Ассоциативные правила могут также служить в качестве эвристических правил, по которым выполняется упреждающая выборка документов, для уменьшения времени ожидания загрузки страниц с удаленного сайта.  
     
   Методы кластеризации в области исследования использования Web-ресурсов применяются как для кластеризации пользователей, так и для кластеризации страниц. Кластеризация пользователей позволяет группировать пользователей с похожим поведением просмотра страниц. Такие знания полезны для того, чтобы сделать выводы о демографии пользователей и выполнении маркетинговой сегментации рынка в электронной коммерции или обеспечении пользователей персональным Web-контентом.  
   Кластеризация страниц позволяет выявить группы страниц с близким по смыслу содержимым. Эта информация полезна для поисковых машин и персональных ассистентов. В обоих случаях пользователям могут предлагаться гиперссылки в соответствии с их запросом и историей запрашиваемой информации.  
     
   Методы классификации могут быть использованы для развития профилей пользователей, относящихся к определенному классу или категории. Это требует построения и выбора функции, которая бы наилучшим образом описывала свойства данного класса. Для этого могут быть использованы любые методы классификации: деревья решений, метод Байеса, SVM и др. Например,  
     
   методы классификации позволяют обнаружить следующее правило: 30 % пользователей, разместивших заказ в разделе "Продукты/Музыка", имеют возраст от 18 до 25 лет и проживают в крупных городах.  
     
   Методы обнаружения шаблонов в последовательностях применяются для выявления межсессионных шаблонов, в которых элементы следуют друг за другом в упорядоченном по времени множестве сессий и эпизодов. Данный подход может помочь в прогнозировании структуры будущих посещений, что в свою очередь помогает в правильном размещении рекламы, нацеленной на конкретную пользовательскую аудиторию. Также методы временного анализа могут быть использованы для анализа трендов, обнаружения точек изменения и анализа подобия.  
     
   Построение модели зависимостей также может широко использоваться применительно к Web. Целью такого анализа является разработка модели, включающей в себя наиболее значимые зависимости между разными переменными в области Web. Так, например, может быть построена модель зависимости между этапами, которые проходит посетитель, и фактом совершения покупки в интернет-магазине (т. е. модель, отличающая случайного посетителя от потенциального покупателя). Существует несколько вероятностных методов обучения модели, которые могут быть использованы для построения модели поведения пользователя при просмотре Web, включая скрытые модели Маркова (Hidden Markov Models) и Байесовские сети доверия (Bayesian Belief Networks).  
     
   Моделирование использования Web-ресурсов позволяет обеспечить не только теоретическую основу поведения пользователей, но и помочь в предсказании. Это способствует увеличению продаж продуктов, размещаемых на сайте, а также улучшению навигации.
2. Этап анализа шаблонов и их применение

Последним этапом в исследовании использования Web-ресурсов является анализ извлеченных шаблонов. Целью анализа является отфильтровать наиболее интересные шаблоны и отбросить ничего не значащие шаблоны. Методология анализа во многом зависит от области применения, в которой он выполняется. Более общей формой анализа шаблонов является механизм запроса знаний, такой как SQL. Другой метод заключается в загрузке данных в куб данных для применения к нему OLAP-операций. Методы визуализации, такие как раскрашивание или графическое изображение шаблонов, могут выделять характерные шаблоны или тренды в данных. Контент или информация о структуре может быть использована для фильтрации страниц, используемых определенным образом, содержащих информацию определенного типа или страницы, имеющие определенную структуру гиперссылок.

В литературе приводится классификация существующих систем анализа использования Web-ресурсов. Классификация выполняется по пяти характеристикам:

- источнику данных: сторона сервера, сторона клиента и прокси;

- типу данных: структуры, контент и информация об использовании;

- количеству пользователей: однопользовательские и многопользовательские;

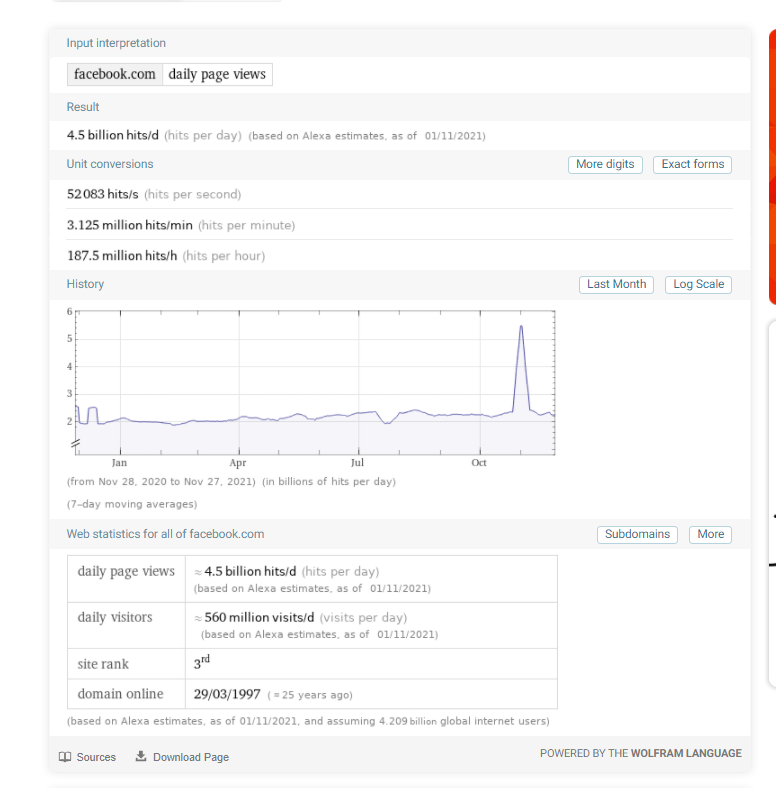
- количеству сайтов: один или множество сайтов;

- области применения.

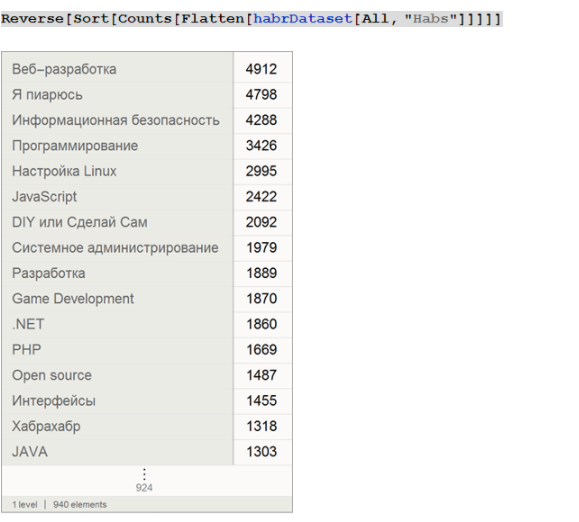
Выделяют следующие области применения систем анализа использования Web-ресурсов.

- Персонализация (Personalization) — обеспечивает для каждого пользователя индивидуальный подход и является одной из важнейших задач для многих Web-систем (например, систем электронной коммерции). Такая персонализация позволяет давать наиболее эффективные рекомендации пользователям в достижении их целей.

- Улучшение систем (System Improvement) — анализ использования Web-ресурсов позволяет рассматривать (выявлять закономерности и взаимосвязи) изменения трафика, обращения к страницам, поведение пользователей. Результаты анализа могут быть применены для разработки политики кэширования, балансировки нагрузки и распределения данных. Это позволяет повысить производительность систем. Кроме того, выявление в закономерностях поведения пользователей позволяет выявлять атаки на сайты и тем самым повышать их безопасность.

- Модификация сайтов (Site Modification) — анализ использования Web-ресурсов обеспечивает дизайнера сайта своего рода обратной связью от пользователей и информацией, необходимой для принятия решения об изменении структуры и его содержания.  
  


- Бизнес-интеллект (Business Intelligence) — выполняет анализ информации об использовании пользователями данных с Web-сайтов, совместно с маркетинговой информацией из электронной коммерции. Выделяют четыре задачи, решаемых в этой области: привлечение новых пользователей, удержание пользователей, проведение перекрестных продаж и определение отказа пользователя от Web-сайта.



# **Выводы**

- Web Mining включает в себя следующие этапы: поиск ресурсов, извлечение информации, обобщение и анализ.

- Различают следующие категории задач Web Mining: извлечение Web-контента, извлечение Web-структур и исследование использования Web-ресурсов.

- Извлечение Web-контента может проводиться в целях информационного поиска и с целью сохранения его в базе данных, также различают неструктурированные и слабоструктурированные Web-документы.

- Для извлечения Web-контента из неструктурированных документов используют модели их представления и методы, заимствованные из Text Mining.

- Извлечение Web-контента для формирования базы данных делится на три подзадачи: моделирование и формирование запросов к Web, извлечение информации и интеграция, создание и реструктуризация Web-сайтов.

- В задаче извлечения Web-структур для представления Web используют направленные и ненаправленные графы.

- В решении задачи извлечения структуры Web используются подходы из

области социальных сетей, библиометрики, ранжирования документов

и т. п.

- В задаче исследования использования Web анализу подвергаются вторичные данные о взаимодействии пользователя с Web: протоколы работы, куки, авторизация и т. п.

- Существуют два основных подхода анализа использования Web-ресурсов: преобразование данных использования Web-сервера в реляционные таблицы до выполнения адаптированных методов Data Mining и использование информации из файла протокола непосредственно, применяя специальные методы предварительной обработки.