Big Data Analytics

tutorialspoint

SIMPLY EASY LEARNING

www.tutorialspoint.com

About the Tutorial

The volume of data that one has to deal has exploded to unimaginable levels in the past decade, and at the same time, the price of data storage has systematically reduced. Private companies and research institutions capture terabytes of data about their users' interactions, business, social media, and also sensors from devices such as mobile phones and automobiles. The challenge of this era is to make sense of this sea of data. This is where **big data analytics** comes into picture.

Big Data Analytics largely involves collecting data from different sources, munge it in a way that it becomes available to be consumed by analysts and finally deliver data products useful to the organization business.

The process of converting large amounts of unstructured raw data, retrieved from different sources to a data product useful for organizations forms the core of Big Data Analytics.

In this tutorial, we will discuss the most fundamental concepts and methods of Big Data Analytics.

Audience

This tutorial has been prepared for software professionals aspiring to learn the basics of Big Data Analytics. Professionals who are into analytics in general may as well use this tutorial to good effect.

Prerequisites

Before you start proceeding with this tutorial, we assume that you have prior exposure to handling huge volumes of unprocessed data at an organizational level.

Through this tutorial, we will develop a mini project to provide exposure to a real-world problem and how to solve it using Big Data Analytics. You can download the necessary files of this project from this link: http://www.tools.tutorialspoint.com/bda/

Copyright & Disclaimer

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

Big Data Analytics – Basics

1. Big Data Analytics – Overview

The volume of data that one has to deal has exploded to unimaginable levels in the past decade, and at the same time, the price of data storage has systematically reduced. Private companies and research institutions capture terabytes of data about their users' interactions, business, social media, and also sensors from devices such as mobile phones and automobiles. The challenge of this era is to make sense of this sea of data. This is where **big data analytics** comes into picture.

Big Data Analytics largely involves collecting data from different sources, munge it in a way that it becomes available to be consumed by analysts and finally deliver data products useful to the organization business.

The process of converting large amounts of unstructured raw data, retrieved from different sources to a data product useful for organizations forms the core of Big Data Analytics.

2. Big Data Analytics – Data Life Cycle

Traditional Data Mining Life Cycle

In order to provide a framework to organize the work needed by an organization and deliver clear insights from Big Data, it’s useful to think of it as a cycle with different stages. It is by no means linear, meaning all the stages are related with each other. This cycle has superficial similarities with the more traditional data mining cycle as described in **CRISP methodology**.

CRISP-DM Methodology

The **CRISP-DM methodology** that stands for Cross Industry Standard Process for Data Mining, is a cycle that describes commonly used approaches that data mining experts use to tackle problems in traditional BI data mining. It is still being used in traditional BI data mining teams.

Take a look at the following illustration. It shows the major stages of the cycle as described by the CRISP-DM methodology and how they are interrelated.

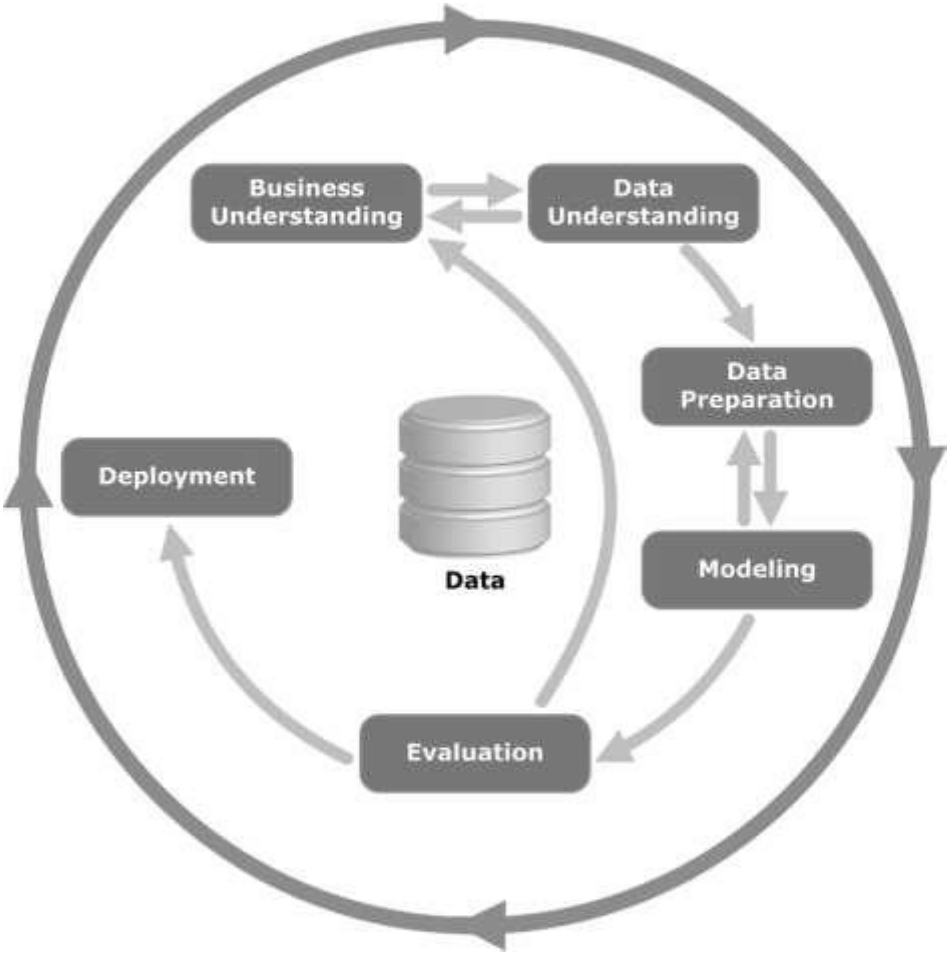


Figure: CRISP-DM life cycle

CRISP-DM was conceived in 1996 and the next year, it got underway as a European Union project under the ESPRIT funding initiative. The project was led by five companies: SPSS, Teradata, Daimler AG, NCR Corporation, and OHRA (an insurance company). The project was finally incorporated into SPSS. The methodology is extremely detailed oriented in how a data mining project should be specified.

Let us now learn a little more on each of the stages involved in the CRISP-DM life cycle:

- **Business Understanding** – This initial phase focuses on understanding the project objectives and requirements from a business perspective, and then converting this knowledge into a data mining problem definition. A preliminary plan is designed to achieve the objectives. A decision model, especially one built using the Decision Model and Notation standard can be used.

- **Data Understanding** – The data understanding phase starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data, or to detect interesting subsets to form hypotheses for hidden information.

- **Data Preparation** – The data preparation phase covers all activities to construct the final dataset (data that will be fed into the modeling tool(s)) from the initial raw data. Data preparation tasks are likely to be performed multiple times, and not in any prescribed order. Tasks include table, record, and attribute selection as well as transformation and cleaning of data for modeling tools.

- **Modeling** – In this phase, various modeling techniques are selected and applied and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem type. Some techniques have specific requirements on the form of data. Therefore, it is often required to step back to the data preparation phase.

- **Evaluation** – At this stage in the project, you have built a model (or models) that appears to have high quality, from a data analysis perspective. Before proceeding to final deployment of the model, it is important to evaluate the model thoroughly and review the steps executed to construct the model, to be certain it properly achieves the business objectives.

  A key objective is to determine if there is some important business issue that has not been sufficiently considered. At the end of this phase, a decision on the use of the data mining results should be reached.

- **Deployment** – Creation of the model is generally not the end of the project. Even if the purpose of the model is to increase knowledge of the data, the knowledge gained will need to be organized and presented in a way that is useful to the customer.

  Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable data scoring (e.g. segment allocation) or data mining process.

In many cases, it will be the customer, not the data analyst, who will carry out the deployment steps. Even if the analyst deploys the model, it is important for the customer to understand upfront the actions which will need to be carried out in order to actually make use of the created models.

SEMMA Methodology

SEMMA is another methodology developed by SAS for data mining modeling. It stands for **S**ample, **E**xplore, **M**odify, **M**odel, and **A**sses. Here is a brief description of its stages:

- **Sample**: The process starts with data sampling, e.g., selecting the dataset for modeling. The dataset should be large enough to contain sufficient information to retrieve, yet small enough to be used efficiently. This phase also deals with data partitioning.

- **Explore**: This phase covers the understanding of the data by discovering anticipated and unanticipated relationships between the variables, and also abnormalities, with the help of data visualization.

- **Modify**: The Modify phase contains methods to select, create and transform variables in preparation for data modeling.

- **Model**: In the Model phase, the focus is on applying various modeling (data mining) techniques on the prepared variables in order to create models that possibly provide the desired outcome.

- **Assess**: The evaluation of the modeling results shows the reliability and usefulness of the created models.

The main difference between CRISM–DM and SEMMA is that SEMMA focuses on the modeling aspect, whereas CRISP-DM gives more importance to stages of the cycle prior to modeling such as understanding the business problem to be solved, understanding and preprocessing the data to be used as input, for example, machine learning algorithms.

Big Data Life Cycle

In today's big data context, the previous approaches are either incomplete or suboptimal. For example, the SEMMA methodology disregards completely data collection and preprocessing of different data sources. These stages normally constitute most of the work in a successful big data project.

A big data analytics cycle can be described by the following stages:

- Business Problem Definition
- Research
- Human Resources Assessment
- Data Acquisition
- Data Munging
- Data Storage
- Exploratory Data Analysis
- Data Preparation for Modeling and Assessment
- Modeling
- Implementation

In this section, we will throw some light on each of these stages of big data life cycle.

Business Problem Definition

This is a point common in traditional BI and big data analytics life cycle. Normally it is a non-trivial stage of a big data project to define the problem and evaluate correctly how much potential gain it may have for an organization. It seems obvious to mention this, but it has to be evaluated what are the expected gains and costs of the project.

Research

Analyze what other companies have done in the same situation. This involves looking for solutions that are reasonable for your company, even though it involves adapting other solutions to the resources and requirements that your company has. In this stage, a methodology for the future stages should be defined.

Human Resources Assessment

Once the problem is defined, it's reasonable to continue analyzing if the current staff is able to complete the project successfully. Traditional BI teams might not be capable to deliver an optimal solution to all the stages, so it should be considered before starting the project if there is a need to outsource a part of the project or hire more people.

Data Acquisition

This section is key in a big data life cycle; it defines which type of profiles would be needed to deliver the resultant data product. Data gathering is a non-trivial step of the process; it normally involves gathering unstructured data from different sources. To give an example, it could involve writing a crawler to retrieve reviews from a website. This involves dealing with text, perhaps in different languages normally requiring a significant amount of time to be completed.

Data Munging

Once the data is retrieved, for example, from the web, it needs to be stored in an easy-to-use format. To continue with the reviews examples, let's assume the data is retrieved from different sites where each has a different display of the data.

Suppose one data source gives reviews in terms of rating in stars, therefore it is possible to read this as a mapping for the response variable $\mathbf{y} \in \mathbf{\{1, 2, 3, 4, 5\}}$. Another data source gives reviews using two arrows system, one for up voting and the other for down voting. This would imply a response variable of the form $\mathbf{y} \in \mathbf{\{positive, negative\}}$.

In order to combine both the data sources, a decision has to be made in order to make these two response representations equivalent. This can involve converting the first data source response representation to the second form, considering one star as negative and five stars as positive. This process often requires a large time allocation to be delivered with good quality.

Data Storage

Once the data is processed, it sometimes needs to be stored in a database. Big data technologies offer plenty of alternatives regarding this point. The most common alternative is using the Hadoop File System for storage that provides users a limited version of SQL, known as HIVE Query Language. This allows most analytics task to be done in similar ways as would be done in traditional BI data warehouses, from the user perspective. Other storage options to be considered are MongoDB, Redis, and SPARK.

This stage of the cycle is related to the human resources knowledge in terms of their abilities to implement different architectures. Modified versions of traditional data warehouses are still being used in large scale applications. For example, teradata and IBM offer SQL databases that can handle terabytes of data; open source solutions such as postgreSQL and MySQL are still being used for large scale applications.

Even though there are differences in how the different storages work in the background, from the client side, most solutions provide a SQL API. Hence having a good understanding of SQL is still a key skill to have for big data analytics.

This stage *a priori* seems to be the most important topic, in practice, this is not true. It is not even an essential stage. It is possible to implement a big data solution that would be working with real-time data, so in this case, we only need to gather data to develop the model and then implement it in real time. So there would not be a need to formally store the data at all.

Exploratory Data Analysis

Once the data has been cleaned and stored in a way that insights can be retrieved from it, the data exploration phase is mandatory. The objective of this stage is to understand the data, this is normally done with statistical techniques and also plotting the data. This is a good stage to evaluate whether the problem definition makes sense or is feasible.

Data Preparation for Modeling and Assessment

This stage involves reshaping the cleaned data retrieved previously and using statistical preprocessing for missing values imputation, outlier detection, normalization, feature extraction and feature selection.

Modeling

The prior stage should have produced several datasets for training and testing, for example, a predictive model. This stage involves trying different models and looking forward to solving the business problem at hand. In practice, it is normally desired that the model would give some insight into the business. Finally, the best model or combination of models is selected evaluating its performance on a left-out dataset.

Implementation

In this stage, the data product developed is implemented in the data pipeline of the company. This involves setting up a validation scheme while the data product is working, in order to track its performance. For example, in the case of implementing a predictive model, this stage would involve applying the model to new data and once the response is available, evaluate the model.

3. Big Data Analytics – Methodology

In terms of methodology, big data analytics differs significantly from the traditional statistical approach of experimental design. Analytics starts with data. Normally we model the data in a way to explain a response. The objectives of this approach is to predict the response behavior or understand how the input variables relate to a response. Normally in statistical experimental designs, an experiment is developed and data is retrieved as a result. This allows to generate data in a way that can be used by a statistical model, where certain assumptions hold such as independence, normality, and randomization.

In big data analytics, we are presented with the data. We cannot design an experiment that fulfills our favorite statistical model. In large-scale applications of analytics, a large amount of work (normally 80% of the effort) is needed just for cleaning the data, so it can be used by a machine learning model.

We don't have a unique methodology to follow in real large-scale applications. Normally once the business problem is defined, a research stage is needed to design the methodology to be used. However general guidelines are relevant to be mentioned and apply to almost all problems.

One of the most important tasks in big data analytics is **statistical modeling**, meaning supervised and unsupervised classification or regression problems. Once the data is cleaned and preprocessed, available for modeling, care should be taken in evaluating different models with reasonable loss metrics and then once the model is implemented, further evaluation and results should be reported. A common pitfall in predictive modeling is to just implement the model and never measure its performance.

4. Big Data Analytics – Core Deliverables

As mentioned in the big data life cycle, the data products that result from developing a big data product are in most of the cases some of the following:

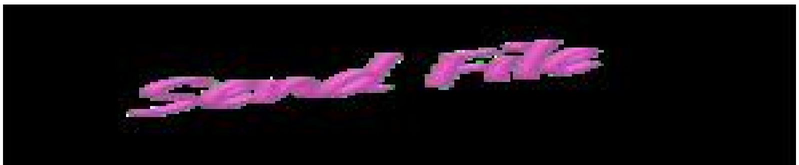
- **Machine learning implementation**: This could be a classification algorithm, a regression model or a segmentation model.
- **Recommender system**: The objective is to develop a system that recommends choices based on user behavior. **Netflix** is the characteristic example of this data product, where based on the ratings of users, other movies are recommended.
- **Dashboard**: Business normally needs tools to visualize aggregated data. A dashboard is a graphical mechanism to make this data accessible.
- **Ad-Hoc analysis**: Normally business areas have questions, hypotheses or myths that can be answered doing ad-hoc analysis with data.

5. Big Data Analytics – Key Stakeholders

In large organizations, in order to successfully develop a big data project, it is needed to have management backing up the project. This normally involves finding a way to show the business advantages of the project. We don't have a unique solution to the problem of finding sponsors for a project, but a few guidelines are given below:

- Check who and where are the sponsors of other projects similar to the one that interests you.
- Having personal contacts in key management positions helps, so any contact can be triggered if the project is promising.
- Who would benefit from your project? Who would be your client once the project is on track?
- Develop a simple, clear, and exiting proposal and share it with the key players in your organization.

The best way to find sponsors for a project is to understand the problem and what would be the resulting data product once it has been implemented. This understanding will give an edge in convincing the management of the importance of the big data project.



6. Big Data Analytics – Data Analyst

A data analyst has reporting-oriented profile, having experience in extracting and analyzing data from traditional data warehouses using SQL. Their tasks are normally either on the side of data storage or in reporting general business results. Data warehousing is by no means simple, it is just different to what a data scientist does.

Many organizations struggle hard to find competent data scientists in the market. It is however a good idea to select prospective data analysts and teach them the relevant skills to become a data scientist. This is by no means a trivial task and would normally involve the person doing a master degree in a quantitative field, but it is definitely a viable option. The basic skills a competent data analyst must have are listed below:

- Business understanding
- SQL programming
- Report design and implementation
- Dashboard development

7. Big Data Analytics – Data Scientist

The role of a data scientist is normally associated with tasks such as predictive modeling, developing segmentation algorithms, recommender systems, A/B testing frameworks and often working with raw unstructured data.

The nature of their work demands a deep understanding of mathematics, applied statistics and programming. There are a few skills common between a data analyst and a data scientist, for example, the ability to query databases. Both analyze data, but the decision of a data scientist can have a greater impact in an organization.

Here is a set of skills a data scientist normally need to have:

- Programming in a statistical package such as: R, Python, SAS, SPSS, or Julia
- Able to clean, extract, and explore data from different sources
- Research, design, and implementation of statistical models
- Deep statistical, mathematical, and computer science knowledge

In big data analytics, people normally confuse the role of a data scientist with that of a data architect. In reality, the difference is quite simple. A data architect defines the tools and the architecture the data would be stored at, whereas a data scientist uses this architecture. Of course, a data scientist should be able to set up new tools if needed for ad-hoc projects, but the infrastructure definition and design should not be a part of his task.

Big Data Analytics – Project

8. Big Data Analytics – Problem Definition

Through this tutorial, we will develop a project. Each subsequent chapter in this tutorial deals with a part of the larger project in the mini-project section. This is thought to be an applied tutorial section that will provide exposure to a real-world problem. In this case, we would start with the problem definition of the project.

Project Description

The objective of this project would be to develop a machine learning model to predict the hourly salary of people using their curriculum vitae (CV) text as input.

Using the framework defined above, it is simple to define the problem. We can define $X = \{x_1, x_2, \ldots, x_n\}$ as the CV's of users, where each feature can be, in the simplest way possible, the amount of times this word appears. Then the response is real valued, we are trying to predict the hourly salary of individuals in dollars.

These two considerations are enough to conclude that the problem presented can be solved with a supervised regression algorithm.

Problem Definition

Problem Definition is probably one of the most complex and heavily neglected stages in the big data analytics pipeline. In order to define the problem a data product would solve, experience is mandatory. Most data scientist aspirants have little or no experience in this stage.

Most big data problems can be categorized in the following ways:

- Supervised classification
- Supervised regression
- Unsupervised learning
- Learning to rank

Let us now learn more about these four concepts.

Supervised Classification

Given a matrix of features $X = \{x_1, x_2, \ldots, x_n\}$ we develop a model M to predict different classes defined as $y = \{c_1, c_2, \ldots, c_n\}$. For example: Given transactional data of customers in an insurance company, it is possible to develop a model that will predict if a client would churn or not. The latter is a binary classification problem, where there are two classes or target variables: churn and not churn.

Other problems involve predicting more than one class, we could be interested in doing digit recognition, therefore the response vector would be defined as: $y = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, a-state-of-the-art model would be convolutional neural network and the matrix of features would be defined as the pixels of the image.

Supervised Regression

In this case, the problem definition is rather similar to the previous example; the difference relies on the response. In a regression problem, the response $y \in \Re$, this means the response is real valued. For example, we can develop a model to predict the hourly salary of individuals given the corpus of their CV.

Unsupervised Learning

Management is often thirsty for new insights. Segmentation models can provide this insight in order for the marketing department to develop products for different segments. A good approach for developing a segmentation model, rather than thinking of algorithms, is to select features that are relevant to the segmentation that is desired.

For example, in a telecommunications company, it is interesting to segment clients by their cellphone usage. This would involve disregarding features that have nothing to do with the segmentation objective and including only those that do. In this case, this would be selecting features as the number of SMS used in a month, the number of inbound and outbound minutes, etc.

Learning to Rank

This problem can be considered as a regression problem, but it has particular characteristics and deserves a separate treatment. The problem involves given a collection of documents we seek to find the most relevant ordering given a query. In order to develop a supervised learning algorithm, it is needed to label how relevant an ordering is, given a query.

It is relevant to note that in order to develop a supervised learning algorithm, it is needed to label the training data. This means that in order to train a model that will, for example, recognize digits from an image, we need to label a significant amount of examples by hand. There are web services that can speed up this process and are commonly used for this task such as amazon mechanical turk. It is proven that learning algorithms improve their performance when provided with more data, so labeling a decent amount of examples is practically mandatory in supervised learning.

9. 9.Big Data Analytics – Data Collection

Data collection plays the most important role in the Big Data cycle. The Internet provides almost unlimited sources of data for a variety of topics. The importance of this area depends on the type of business, but traditional industries can acquire a diverse source of external data and combine those with their transactional data.

For example, let's assume we would like to build a system that recommends restaurants. The first step would be to gather data, in this case, reviews of restaurants from different websites and store them in a database. As we are interested in raw text, and would use that for analytics, it is not that relevant where the data for developing the model would be stored. This may sound contradictory with the big data main technologies, but in order to implement a big data application, we simply need to make it work in real time.

Twitter Mini Project

Once the problem is defined, the following stage is to collect the data. The following mini-project idea is to work on collecting data from the web and structuring it to be used in a machine learning model. We will collect some tweets from the twitter rest API using the R programming language.

First of all create a twitter account, and then follow the instructions in the **twitteR** package vignette to create a twitter developer account. This is a summary of those instructions:

- Go to **https://twitter.com/apps/new** and log in.
- After filling in the basic info, go to the "Settings" tab and select "Read, Write and Access direct messages"
- Make sure to click on the save button after doing this
- In the "Details" tab, take note of your consumer key and consumer secret
- In your R session, you'll be using the API key and API secret values
- Finally run the following script. This will install the **twitteR** package from its repository on github

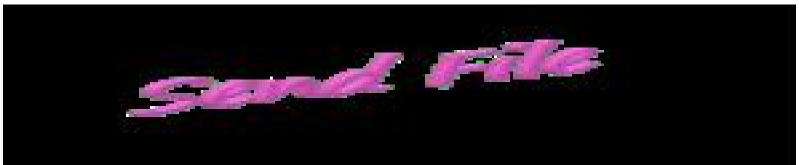
```
install.packages(c("devtools", "rjson", "bit64", "httr"))



# Make sure to restart your R session at this point

library(devtools)

install_github("geoffjentry/twitteR")
```



We are interested in getting data where the string "big mac" is included and finding out which topics stand out about this. In order to do this, the first step is collecting the data from twitter. Below is our R script to collect required data from twitter. This code is also available in bda/part1/collect_data/collect_data_twitter.R file.

```
rm(list = ls(all = TRUE)); gc() # Clears the global environment



library(twitteR)

Sys.setlocale(category = "LC_ALL", locale = "C")



### Replace the xxx’s with the values you got from the previous instructions

# consumer_key = "xxxxxxxxxxxxxxxxxxxx"

# consumer_secret = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"

# access_token = "xxxxxxxxxx-xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"

# access_token_secret= "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"



# Connect to twitter rest API

setup_twitter_oauth(consumer_key, consumer_secret, access_token, access_token_secret)



# Get tweets related to big mac

tweets <- searchTwitter(’big mac’, n=200, lang=’en’)

df <- twListToDF(tweets)



# Take a look at the data

head(df)



# Check which device is most used

sources <- sapply(tweets, function(x) x$getStatusSource())

sources <- gsub("</a>", "", sources)

sources <- strsplit(sources, ">")
sources <- sapply(sources, function(x) ifelse(length(x) > 1, x[2], x[1]))

source_table = table(sources)

source_table = source_table[source_table > 1]

freq = source_table[order(source_table, decreasing=T)]

as.data.frame(freq)


#                       Frequency

# Twitter for iPhone       71

# Twitter for Android      29

# Twitter Web Client       25

# recognia                 20
```

10. Big Data Analytics – Cleansing Data

Once the data is collected, we normally have diverse data sources with different characteristics. The most immediate step would be to make these data sources homogeneous and continue to develop our data product. However, it depends on the type of data. We should ask ourselves if it is practical to homogenize the data.

Maybe the data sources are completely different, and the information loss will be large if the sources would be homogenized. In this case, we can think of alternatives. Can one data source help me build a regression model and the other one a classification model? Is it possible to work with the heterogeneity on our advantage rather than just lose information? Taking these decisions are what make analytics interesting and challenging.

In the case of reviews, it is possible to have a language for each data source. Again, we have two choices:

- **Homogenization**: It involves translating different languages to the language where we have more data. The quality of translations services is acceptable, but if we would like to translate massive amounts of data with an API, the cost would be significant. There are software tools available for this task, but that would be costly too.

- **Heterogenization**: Would it be possible to develop a solution for each language? As it is simple to detect the language of a corpus, we could develop a recommender for each language. This would involve more work in terms of tuning each recommender according to the amount of languages available but is definitely a viable option if we have a few languages available.

Twitter Mini Project

In the present case we need to first clean the unstructured data and then convert it to a data matrix in order to apply topics modelling on it. In general, when getting data from twitter, there are several characters we are not interested in using, at least in the first stage of the data cleansing process.

For example, after getting the tweets we get these strange characters: "<ed><U+00A0><U+00BD><ed><U+00B8><U+008B>". These are probably emoticons, so in order to clean the data, we will just remove them using the following script. This code is also available in bda/part1/collect_data/cleaning_data.R file.

```
rm(list = ls(all = TRUE)); gc() # Clears the global environment

source('collect_data_twitter.R')



# Some tweets

head(df$text)

[1] "I’m not a big fan of turkey but baked Mac &amp; cheese
<ed><U+00A0><U+00BD><ed><U+00B8><U+008B>"
```

```
[2] "@Jayoh30 Like no special sauce on a big mac. HOW"
### We are interested in the text - Let’s clean it!

# We first convert the encoding of the text from latin1 to ASCII
df$text <- sapply(df$text,function(row) iconv(row, "latin1", "ASCII", sub=""))

# Create a function to clean tweets
clean.text <- function(tx){
  tx <- gsub("htt.{1,20}", " ", tx, ignore.case=TRUE)
  tx = gsub("[^#[:^punct:]]|@|RT", " ", tx, perl=TRUE, ignore.case=TRUE)
  tx = gsub("[[:digit:]]", " ", tx, ignore.case=TRUE)
  tx = gsub(" {1,}", " ", tx, ignore.case=TRUE)
  tx = gsub("^\\s+|\\s+$", " ", tx, ignore.case=TRUE)
  return(tx)
}

clean_tweets <- lapply(df$text, clean.text)

# Cleaned tweets
head(clean_tweets)
[1] " WeNeedFeminlsm MAC s new make up line features men woc and big girls "
[1] " TravelsPhoto What Happens To Your Body One Hour After A Big Mac "
```