ISSN: 0976-3031

Available Online at http://www.recentscientific.com

CODEN: IJRSFP (USA)

International Journal of Recent Scientific Research
Vol. 10, Issue, 12(A), pp. 36216-36224, December, 2019

International Journal of Recent Scientific Research

DOI: 10.24327/IJRSR

Research Article

BIG DATA ANALYSIS TOOLS COMBINED WITH AHP FOR IMPROVING BANK SERVICES SALES

Mayor-Vitoria F[1], Garcia-Bernabeu A[2], Pla-Santamaria D[3], Salas-Molina F[4] and Nor Aida Abdul Rahman[5]

[1,2,3]Universitat Politècnica de València
[4]Universidad de Valencia
[5]Universiti Kuala Lumpur

DOI: http://dx.doi.org/10.24327/ijrsr.2019.1012.4882

ARTICLE INFO

Article History:

Received 13th September, 2019
Received in revised form 11th October, 2019
Accepted 8th November, 2019
Published online 28th December, 2019

Key Words:

Analytic Hierarchy Process, decision-making, client selection, research priorities, sales professionals, artificial intelligence.

ABSTRACT

This paper deals with two important issues related to the decision making in the financial field: Big Data and Multicriteria Decision Making (MCDM) methods. To handle the combination between them, we apply the so-called MapReduce paradigm, which is widely deployed in big data analysis, and the Analytic Hierarchy Process (AHP), which is the most used method among the MCDM methodologies. The main gap to cover is shown in two directions; on the one hand, how big data analysis can help to overcome the limitations of methodologies such as AHP when a vast number of alternatives are present, on the other hand, we look at how MCDM methods can help big data analysis to go one step beyond, that is to say, to move from the predictive to the prescriptive analysis. To illustrate the whole approach, we show its application to a real world decision problem concerning the sale of travel insurances. Our methodology returns an accurate ranking of potential clients before being contacted by the sales agent working for a commercial bank. So it helps to the sales profession by contributing to the creation of value for customers and to the sales professionals by optimizing their functions.

INTRODUCTION

In recent years, the treatment of information has become crucial for public and private decision makers of any kind of business or public institution. In fact, how the information is obtained, stored and managed has always been important, but nowadays, the main difference lies in the huge quantity of data obtained every single minute and how fast they are generated. According to Menon (2014), only Facebook can store more than 500 terabytes of photos and videos every day. Taking into account that there are other very popular social media networks like Twitter, Google+ or apps such as Whatsapp which are used worldwide, the magnitude of stored data is something that escapes the human understanding. For instance, Google processes more than 24 petabytes of data per day and 400 million tweets are sent per day worldwide, see John Walker (2014). These amount of data can be used to approach to new customers, in fact, regarding the sales perspective, Andzulis *et al*. (2012) already reviews the role of social media in the sales force and the sales process. Also, Marshall *et al*. (2012) describe the influence of social media as a revolution in the buyer-seller relationship.

The management of huge quantities of information has been previously solved by the implementation of relational databases, Sapna (2017), which are composed by a large but limited number of tables connected to each other. However, these systems present some limitations in terms of the quantity of data to be stored and the speed access to them, see Baby *et al*. (2016). Therefore, new goals are related to storing and managing those huge volumes of data which are complex and/or unstructured, but in a reasonable time. Moreover, in traditional database systems, data are stored in order to be used and processed according to a specific purpose. However, due to the cost reduction in storage systems, the trend nowadays is to save the information even if this is not significant currently. In fact, IBM SPSS statistics guide to data analysis Norusis (2011) estimates that 90% of world's data have been created since 2011. In addition, the trend is that the gap between unstructured and structured data will increase until 2020, see George and Mallery (2016)

Corresponding author:* **Mayor-Vitoria F
Universitat Politècnica de València

Big data analytics appears as a solution to solve the problem of how to process the big volume of unstructured information generated by humans and machines every day because as LaValle *et al* (2011) explained, sophistication in the information analysis is basic to get high levels of performance in companies. In this context, Singh *et al*. (2019) reveals that the influence of sales digitalization technologies, which include digitization and artificial intelligence, is likely to be more significant and more far reaching than previous sales technologies. Moreover, as explained in Ramanathan *et al*. (2017), analytics can be grouped into three categories: a) descriptive analytics (captures what happened, see Underhill (2009)), b) predictive analytics (it aims to predict the future, see Hays (2004)) and c) prescriptive analytics (it provides the best alternative to solve the problem, see Kant *et al*. (2008) and Mahadevan *et al*. (2013)). Another gap here is that although the added value that prescriptive analytics can give to a company, only a very small number of organizations use it and in this context Multiple Criteria Decision Making (MCDM) appears to be a solution to help human mind to achieve the skill to make the right decision in a complex environment and with limited time for decision making, see Simon (1972)

MCDM helps Decision Makers in solving choice, ranking and sorting out problems concerning a set of alternatives evaluated on multiple criteria, see Contini and Zionts (1968). MCDM is a branch of Operations Research models that started to emerge in the 1950s. However, it had not been an active area of research until the 1970s with important contributions from Zionts and Wallenius (1976) or Xidonas *et al*. (2009). Saaty introduced in 1977, see Saaty (1977), the Analytic Hierarchy Process (AHP), a multicriteria method that relies on pairwise comparison of criteria/assets to be evaluated from the decision maker's preferences. Indeed, one of the objectives of this paper is to combine big data analytics with AHP to compare several criteria which describe the habits of the different alternatives, in other words, to analyze the way of buying of the clients of a bank or a finance institution.

The process of applying the AHP's algorithm and the procedure to obtain the final ranking is explained in Section 4, where a detailed case study involving four criteria and 3000 alternatives is carried out. The main reason why the Analytic Hierarchy Process (AHP) is used in this study is that the AHP is a flexible and intuitive method for decision makers, which also calculates the consistency of the judgments of the experts, which in our case are the bank managers. Several studies demonstrate the relevance of AHP as a subjective weighting procedure to obtain weights at individual level by a straightforward approach, see Kwong and Bai (2002). In addition, the flexibility of AHP and its simplicity to help the decision maker to prioritize criteria and alternatives have significant advantages over other subjective weighting methods, see Maggino and Ruviglioni (2009). In our case, the pairwise comparison of the four criteria is made by one expert, the bank branch manager. But the pairwise comparison among the 3000 alternatives for the four criteria is in a very accurate way thanks to the previous big data analysis in which exact values for each criterion are extracted from a vast amount of unstructured information given by the cards transactions.

The purpose of the present paper is: (a) To explain how a previous big data analysis can help MCDM to be more efficient, (b) To generate a MapReduce code in Python in order to extract relevant information from an unstructured database containing the transactions of all clients cards for a period; (c) To apply the AHP methodology in order to provide the big data analysis with an added value in terms of a more prescriptive framework; and (d) To illustrate the aforementioned by the implementation of a case study to offer personal travel insurances to real potential clients.

This paper is organized as follows. In section 2, we introduce some basic concepts relative to big data analytics. In section 3, the proposed methodology combining MapReduce and AHP is presented. A real world multicriteria problem, related to the raking of bank clients, illustrates the considered methodology in section 4. Conclusions are drawn and some future directions of research are provided in Section 5.

Basic Concepts about Big Data Analytics

Neither the origins nor the big data definition are clear at all, so both still involve some confusion. In fact, there is no standard definition agreed upon by the scientific community and this is why the term involves some subjectivity. In general, most of the definitions have in common the so-called 3Vs (volume, velocity and variety) suggested by Beyer and Laney (2012) and used by many authors as the most widely used properties to describe it, such as Chen *et al*. (2012).. Moreover, maybe the real origin of the term was figured out by John Mashey during an informal conversation at Silicon Graphics Inc in the 1990s, as is commented in Diebold (2012).

Normally, the term big data involves a huge amount of information to be processed, but it is worth highlighting the point at which all three dimensions are equally important when talking about big data. In fact, some companies can use big data to analyze not a big volume of information, but having great results. For example, Gartner (2017) defines big data by positioning all the dimensions at the same level: "Big data is high-volume, high-velocity and high-variety information assets that demand cost-effective, innovative forms of information processing for enhanced insight and decision making".

In addition to the three main characteristics above mentioned, four more Vs have been included to describe big data more precisely. These include: veracity, variability, visualization and value. Hereafter, the seven dimensions are described:

-Volume: It refers to the quantity of data which is generated in order to process them so that they are converted into actions. For instance, social networks generate petabytes of data every day. Beaver *et al*. (2010) reported that Facebook processes more than one million photographs per second.

- Velocity: This is how fast data are created, stored and processed in real time. According to Cukier and Mayer-Schoenberger (2013), Wal-Mart can process more than one million transactions per hour.
- Variety: It shows the type of data, such as, text documents, emails, social media profiles, audios, videos or images. All these are examples of unstructured, which represents 95% of all existing data John Walker (2014).
- Veracity: It means the grade of reliability of the data received.

- Variability: It includes, flow rates of data can have peaks and troughs as velocity is not always stable.
- Visualization: In which way data are displayed in order to find relevant information for the user.
- Value: When data are converted into knowledge and then, into valuable information for decision makers (DM).

Due to its popularity and its relevance in data generation, social media networks are often used as examples of the massive unstructured data generators. However, it is not the only source of information. Herewith, the most important sources of data are presented:

- Web and Social Media: It includes web content and data obtained from social networks such as Facebook, Twitter, LinkedIn, etc, blogs.
- Machine-to-Machine (M2M): It refers to technologies which allow the connection with other gadgets. M2M uses gadgets such as sensors which capture some event in particular. Examples of those events are speed, temperature, pressure, weather or chemical variables among others. All these are transmitted to other apps that transform these data into relevant information.
- Big Transaction Data: It includes registers such as purchasing transactions or call logs in telecommunications. It includes billing records and detailed telecommunications call records. These transactional data are available in both semi-structured and unstructured formats.
- Biometrics: This means information that includes fingerprints, retinal scanning, facial recognition or genetics. In the area of security and intelligence, biometric data have been important information for research agencies
- Human Generated: Humans also generate several quantities of data like the information stored by a call center when a phone call is stabilized, audio notes, emails, electronic documents, medical reports and many other daily life actions.

Moreover, the expanding use of new technologies such as Internet of the Things (IoT) makes any gadget used nowadays generate data which can be analyzed, see Atzori *et al*.

(2010). From all these data, only a small portion is considered structured while the vast majority is unstructured. Therefore, big data appears as a solution to solve both problems: a) how information is stored and b) how to get value from these diverse data. Hereafter, the main components of big data will be explained in order to understand how it works internally.

It has been mentioned that the management done by traditional relational database systems is strict because all new data need to conform to a scheme and then, any minor change will involve altering the current records and it will generate additional costs, see Han *et al*. (2011). Hence, a flexible structure is required in order to introduce new real-world object forms dynamically, but also another challenge for relational database systems is scalability when billions of records have to be stored, see Agrawal *et al*. (2011). In this context, the NoSQL (Not only SQL) systems appear as a solution to read and save huge volume of data per second by scaling out systems. In case a table is too big to be stored in a single system, NoSQL divides and splits it into several tables distributed in different servers and then large-scale data ingestions are supported.

There are several technologies which allow organizations to work with big data, however, Apache Hadoop's, which was created by Doug Cutting and Mike Cafarella in 2005, has become the most popular among the software community. The Hadoop ecosystem is complex, but it is possible to distinguish three main layers, see Figure 1. The bottom layer is the distributed file system which is prepared to store and scale Hexabytes of data. The following layer is YARN (Yet Another Resource Negotiator), which takes care of the number of systems found in a cluster, the resources consumed and how the different operations are scheduled. And the final one is the layer dedicated to the MapRefuce paradigm which allows the execution of tasks in parallel on the nodes which contain the data.

Once large volumes of data have been stored, the next step consists in processing and analyzing them. Since these data are large and they are stored among multiple systems, the program needs to be introduced into the machine that contains the data. So the old protocols based on: "data to the process" are moved to: "process to data". MapReduce programming protocol provides this characteristic because the code program is executed independently of whether the data are stored in one single system or in a system formed by multiple clusters.

The MapReduce paradigm consists in two parts basically: on one side, the Mapper and on the other side, the Reducer. Depending on how complex the subject matter is, more than one Mapper can be used and also more than one Reducer, but in any case, the procedure to work is always the same. In Figure 2, a simple word count task using MapReduce is shown as an example.

In general, the Mapper processes the unstructured data so the Reducers can report useful results to obtain significant information. On the one hand, the Mapper searches among all the unstructured information and creates groups through a particular key. The key is the element for which the Mapper sorts and shuffles the information. Therefore, what Mappers do, consist in searching among the different keys which can exist in a unstructured database and group them, so it will be easier to get values from each key. On the other hand, the Reducer is more complex, and the degree of complexity depends on the application to deal with. Normally the Reducer takes the information generated by the Mapper and reports results with relevant information for the case study. Reducers do so by going through the different lines generated by the Mapper and analyzing them. From each one of the lines, it can extract information which can be reported or even stored to be processed later by the same Reducer. In any case, Reducer will always provide a final result for the user who has executed it.

METHODOLOGY

The working methodology is shown in Figure 3 and the procedure is divided in two main blocks. Both parts are connected between them and they report valuable information to each other: Big data analytics and the MCDM procedure.

First, the problem is defined after studying the bank's requirements which help us to know the objective which is set

on the top of the AHP diagram. Also, after analyzing the problem, we establish a serie of criteria and alternatives. In the case study, see Section 4, four criteria are defined and justified for a real decision making problem. There is a component of human interpretation when the comparison about the importance of each criterion is made. Regarding the alternatives, these will be some clients from the finance institution, but the number is uncertain at this point of the process because the information comes from an unstructured database which is not possible to filter by a traditional system.
Knowing the criteria to be considered and the type (not the quantity) of alternatives we want to find and evaluate, the next stage consists in programming a Mapper and a Reducer program for each criterion. The code program for each Mapper and Reducer is written in Python language and the core code for each one is shown in Section 4. We have developed one Mapper and one Reducer for each criterion.

The result of each MapReduce program is the analytic report which will tell us the values of each criterion for each alternative. This exact and accurate information is sent to the AHP system and it is necessary to create the comparison tables of each criterion per alternative. Moreover, the analytic report will inform about the number of alternatives to be considered.
With all the above information, the AHP can be executed and then we can get the final ranking with the top potential clients to be contacted by the sales agents of the finance institution.

CASE STUDY AND RESULTS

Following the methodology described in Section 3, we have conducted a real example over one million registered transactions made by thousands of different clients of a bank in order to illustrate all mentioned in previous sections. Thus, big data properties are enhanced because of the amount of data and unstructured information. In fact, the txt file used is over 60MB. In the financial field, also huge quantities of data are generated every single minute when the bank's clients make transactions. Payments by credit or debit cards, operations in an ATM or transactions between private users can generate a massive amount of unstructured data. If these data are analyzed properly, they can provide the issuing entity with valuable information related to the lifestyle of their clients and therefore, the possibility of offering them the most suitable services in accordance to their needs.

In this case, the problem definition is clear: to sell personal travel insurances by phone. This is considered a complicated task, especially when the sales agent does not have a proper understanding of what the potential client really needs. In this sense, we consider that the objective which will be on the top of our AHP is: the clients' selection, that is to say, a list with the top potential clients to be contacted.

Also, from the problem definition we can deduce that the alternatives will be all those clients who have made a transaction with the credit or debit card during the considered period, but as the database is huge and unstructured, we cannot know how many clients are present at the database at that moment. What we can define after studying the problem are the criteria. These criteria are: a) The purchase concept (C1); b) Road transport (C2); c) Card type (C3) and d) Country of purchasing (C4). All these four criteria are described below and they will report information about how the clients are buying in order to identify their habits in terms of how frequent travelers they are.

- Purchase concept (trip): This is the first criterion and it contains the number of times that a Client has bought an airline ticket, a train ticket or has paid for a hotel reservation. The higher this criterion, the more frequent a traveler is. So this information will tell us about the behavior of the Client in terms of travelling.
- Road Transport: This second criterion will inform about the number of times that the Client has paid for a taxi, an Uber or they have refueled their car's tank. The higher this criterion, the longer on the road they are. So this information is important because those travelling by car have a higher risk of having an accident on the road.
- Type of card: This is the third criterion and it explains if payments are made with a debit card. Again, the higher this criterion, the more purchases using a debit card have been made. This information must be known because those trips paid by debit card are not covered by any travel insurance.
- Place of purchase: The last criterion will inform about the number of purchases that each Client has made outside Spain, so a high number will indicate that the Client is a frequent traveler too.

Once we know the criteria and the type of alternatives we want to find in the database, the next step consists in programming the MapReduce code in Python language. We have chosen Python because it is one of the most accepted languages by Hadoop and compared to others, it is flexible and easy to use. As it was explained in Section 2, every MapReduce code has two parts: the Mapper and the Reducer. In this case, we need four Mappers and four Reducers because we are working with four criteria. Hereafter, the code of each one is shown and explained.

The following Python code corresponds to Mapper for the first criterion (C1). As we know from the bank, each card transaction generates one line in the database, so the first thing we need from the Mapper is to go through all the lines looking for the Key (client's name) and the concept of purchasing.

```
import sys

for line in sys.stdin:
data = line.strip () .split("\ t ")
if len(data) == 7:
date, time, name, concept, cost, payment, country = data
print "{0}\t{1}".format(name , concept)
```

The function line.strip().slip() generates the data array with seven positions. We learnt from the finance institution that normally the cards transactions generate information about seven fields in the following order: date, time, name, concept, cost, payment and country. Therefore, we create and define the array, called data, in the same way. Every time a line is checked, the function len() checks if the array has seven positions. If it does, the third and the fourth position of the line are printed; if does not, the line is dismissed. This loop is repeated for every line in the database and the reported information is a list like this:

Key1, Value

Key1, Value
key1, Value
..
key2, Value
Key2, Value
Key2, Value
..
KeyN, Value
KeyN, Value

Where, Key will be the name of the Client and Value what the client has paid with the credit or debit card.

The following Python code corresponds to the Reducer for the first criterion (C1). Basically, the Reducer has three main steps for each line after taking the list of two elements per line (Key and Value) generated by the Mapper:

a. First, it checks if the data mapped in each line has two elements. If it does not, it avoids that line.
b. If it does, it checks if the Key is the same as the line before, if not, it prints the Key and the value of the variable ”counttrip” and it restarts ”counttrip” to zero.
c. If it does, it goes for the following line and it checks if the concept of purchase is an air ticket, a train ticket or a hotel reservation. If it does, it increases the variable ”counttrip” by one and goes for the following line.

```
import sys

counttrip = 0
oldKey = None

for line in sys.stdin:
data_mapped = line.strip ( ).split("\t ")
if  len (data_mapped) != 2 :
continue

this Key, this Item = data_mapped

if  old Key and old Key != thisKey:
print old Key, "\ t ", counttrip
counttrip = 0

old Key = this Key
if thisItem == 'Avion' or thisItem == 'Tren' or thisItem == 'Hotel':
counttrip += 1
```

The rest of the Mappers and the Reducers are similar to the ones programmed for the C1. In what follows, all of them are shown and explained. The following Python code corresponds to Mapper for the second criterion (C2). As we know from the bank, each card transaction generates one line in the database, so the first thing we need from the Mapper is to go through all the lines looking for the Key (client’s name) and the concept of purchasing as in C1.

```
import sys

for line in sys.stdin:
data = line.strip().split ("\ t ")
if len(data) == 7 :
date, time, name, concept, cost, payment, country = data
print "{0}\ t{1}".format(name, concept)
```

The function line.strip().slip() generates the data array with seven positions. We learnt from the finance institution that normally the cards transactions generate information about seven fields in the following order: date, time, name, concept, cost, payment and country. So we create and define the array, called data, in the same way. Every time a line is checked, the function len() checks if the array has seven positions, if it does, the third and the fourth positions of the line are printed; if it does not, the line is dismissed. This loop is repeated for every line in the database and the reported information is a list like this:

Key1, Value
Key1, Value
Key1, Value
..
Key2, Value
Key2, Value
Key2, Value
..
KeyN, Value
KeyN, Value

Where, Key will be the name of the Client and Value what the Client has paid with the credit or debit card.
The following Python code corresponds to the Reducer for the second criterion (C2). Basically, the Reducer has three main steps for each line after taking the list of two elements per line (Key and Value) generated by the Mapper:

```
import sys

countroad = 0
oldKey = None

for line in sys.stdin:
data_mapped = line.strip().split("\t ")
if  len(data_mapped) != 2:
continue

thisKey , thisItem = data_mapped

if  oldKey and oldKey != thisKey:
print oldKey, "\ t ", countroad
countroad = 0
old Key = thisKey
if thisItem == 'Uber ' or thisItem == 'Taxi ' or thisItem == 'Gasolina':
countroad += 1
```

a. First, it checks if the data mapped in each line has two elements. If it does not, it avoids that line.
b. If it does, it checks if the Key is the same as the line before, if not, it prints the Key and the value of the variable ”countroad” and it restarts ”countroad” to zero.
c. If it does, it goes for the following line and it checks if the concept of purchase is taxi, an Uber or a payment in a petrol station. If so, it increases the variable ”countroad” by one and goes for the following line.

The following Python code corresponds to Mapper for the third criterion (C3). As we know from the bank, each card transaction generates on line in the database, so the first thing we need from the Mapper is to go through all the lines looking for the Key (Client’s name) and and the type of card used.

```
import sys

for line in sys.stdin:
data = line.strip().split("\ t ")
if len(data) == 7:
```

```
date , time , name , concept , cost , payment , country = data
print "{0}\ t{1}".format(name, payment)
```

The function line.strip().slip() generates the data array with seven positions. We learnt from the finance institution that normally the cards transactions generate information about seven fields in the following order: date, time, name, concept, cost, payment and country. So we create and define the array, called data, in the same way. Every time a line is checked, the function len() checks if the array has seven positions, if it does, the third and the sixth position of the line are printed; if it does not, the line is dismissed. This loop is repeated for every line in the database and the reported information is a list like this:

```
Key1, Value
Key1, Value
Key1, Value
..
key2, Value
Key2, Value
Key2, Value
..
KeyN, Value
KeyN, Value
```

Where, Key will be the name of the Client and Value the name of the credit or debit card.

The following Python code corresponds to the Reducer for the third criterion (C3). Basically, the Reducer has three main steps for each line after taking the list of two elements per line (Key and Value) generated by the Mapper:

```
import sys

countcard = 0
oldKey = None

for line in sys.stdin:
data_mapped = line.strip ().split("\t ")
if len(data_mapped) != 2:
continue

this Key, this Item = data_mapped

if old Key and oldKey != thisKey:
print old Key, "\ t " , countcard
countcard = 0

old Key = thisKey
if this Item == 'Electron' or thisItem == 'Maestro':
count card += 1
```

a. First, it checks if the data mapped in each line has two elements. If not, it avoids that line.
b. If it does, it checks if the Key is the same as the line before, if not, it prints the Key and the value of the variable ”countcard” and it restarts ”countcard” to zero.
c. If it does, it goes for the following line and it checks if the concept of purchase is Electron or Maestro. If so, it increases the variable ”countcard” by one and goes for the following line.

Finally, the following Python code corresponds to Mapper for the fourth criterion (C4). As we know from the bank, each card transaction generates one line in the database, so the first thing we need from the Mapper is to go through all the lines looking for the Key (Client’s name) and country where the transaction has been made.

```
import sys

for line in sys.stdin:
data = line.strip().split("\ t ")
if len(data) == 7:
date , time , name , concept , cost , payment , country = data
print "{0}\ t{1}".format(name, country)
```

The function line.strip().slip() generates the data array with seven positions. We learnt from the finance institution that normally the cards transactions generates information about seven fields in the following order: date, time, name, concept, cost, payment and country. So we create and define the array, called data, in the same way. Every time a line is checked, the function len() checks if the array has seven positions, if it does, the third and the seventh position of the line are printed; if it does not, the line is dismissed. This loop is repeated for every line in the database and the reported information is a list like this:

```
Key1, Value
Key1, Value
key1, Value
..
key2, Value
Key2, Value
Key2, Value
..
KeyN, Value
KeyN, Value
```

Where, Key will be the name of the Client and Value the name of the country where the Client has paid.

The following Python code corresponds to the Reducer for the fourth criterion (C4). Basically, the Reducer has three main steps for each line after taking the list of two elements per line (Key and Value) generated by the Mapper:

```
import sys

count country = 0
old Key = None

for line in sys.stdin:
data_mapped = line.strip ().split("\t ")
if len(data_mapped) != 2:
continue

thisKey, thisItem = data_mapped

if oldKey and oldKey != thisKey:
print oldKey, "\ t ", countcountry
countcard = 0

old Key = this Key
if this Item != 'Spain':
count country += 1
```

a. First, it checks if the data mapped in each line has two elements. If not, it avoids that line.
b. If it does, it checks if the Key is the same as the line before, if not, it prints the Key and the value of the variable ”count country” and it restarts ”count country” to zero.
c. If it does, it goes for the following line and it checks if the country of purchasing is Spain. If not, it increased the variable ”count country” by one and goes for the following line.

After executing all the MapReduce programs, the big data analytic report is completed. A total of 3000 clients, from an unstructured database with one million transactions, have been identified and the values of each one for each criterion are shown in Table 1.

Table 1 Values of each criterion per Client - Source: Unstructured database

Client	C1-Trip	C2-Roac	C3-Carc	C4-Country
0001Client	21	8	150	139
0002Client	37	19	133	319
0003Client	43	18	135	179
0004Client	9	28	137	79
0005Client	3	26	141	181
0006Client	12	2	143	69
0007Client	30	22	145	189
0008Client	45	63	145	48
0009Client	28	56	147	203
0010Client	46	5	145	84
0011Client	45	42	141	69
0012Client	9	38	123	51
0013Client	9	8	115	220
0014Client	18	31	139	180
0015Client	29	63	137	37
0016Client	38	22	137	177
0017Client	29	13	124	41
0018Client	6	51	126	56
0019Client	28	43	153	239
0020Client	9	2	135	315
...	...	...	...	...
3000Client	35	45	146	201

These results are introduced in the AHP system developed in Excel together with the pairwise comparison of the criteria, see Table 2 (Criteria matrix comparison or Matrix C), made by the office bank director (human interpretation) according to Saaty's scale:

Table 2 Matrix C - Source: Criteria comparison made by the bank office director's

C	C1	C2	C3	C4	C1-n	C2-n	C3-n	C4-n	Av
C1	1	0.2	0.3333	0.1429	0.0625	0.1154	0.0357	0.0225	0.0590
C2	5	1	3	5	0.3125	0.5769	0.3214	0.7883	0.4998
C3	3	0.3333	1	0.2	0.1875	0.1923	0.1071	0.0315	0.1296
C4	7	0.2	5	1	0.4375	0.1154	0.5357	0.1577	0.3116
SUM	16	17.333	93.333	63.429					

- Criterion 1 (Contept Trip) is: 5 times more important than criterion 2; 3 times more important than criterion 3 and 7 times more important than criterion 4.
- Criterion 3 (Debit Card) is: 3 times more important than criterion 2 and 5 times more important than criterion 4.
- Criterion 4 (Country) is: 5 times more important than criterion 2.

Once all the results above mentioned are introduced in the AHP, this creates the pairwise comparison of the criteria for each alternative according to the Tables 3, 4, 5 and 6.

Table 3 Part of criteria table C1 - Source: Criteria table comparing 3000 clients for C1

C1	0001 Client	0002 Client	0003 Client	0004 Client	0005 Client	0006 Client	0007 Client	Normalized	Av
0001 Client	1	1.7619	2.0476	0.4285	0.1428	0.5714	1.4285	...	0.0014
0002 Client	0.5675	1	1.1621	0.2432	0.081	0.3243	0.8108	...	0.0008
0003 Client	0.4883	0.8604	1	0.2093	0.0697	0.2790	0.6976	...	0.0007
0004 Client	2.3333	4.1111	4.7777	1	0.3333	1.3333	3.3333	...	0.0034
0005 Client	7	1.2333	1.4333	3	1	4	10	...	0.0103
0006 Client	1.75	3.0833	3.5833	0.75	0.25	1	2.5	...	0.0025
0007 Client	0.7	1.2333	1.4333	0.3	0.1	0.4	1	...	0.0010
...	...	...	...	...	...	...	...	...	...
3000 Client	0.6	1.0571	1.2285	0.2571	0.0857	0.3428	0.8571	...	0.0008

Table 4 Part of criteria table C2 - Source: Criteria table comparing 3000 clients for C2

C2	0001 Client	0002 Client	0003 Client	0004 Client	0005 Client	0006 Client	0007 Client	Normalized	Av
0001 Client	1	2.375	2.25	3.5	3.25	0.25	2.75	...	0.0041
0002 Client	0.4210	1	0.9473	1.4736	1.3684	0.1052	1.1578	...	0.0017
0003 Client	0.4444	1.0555	1	1.5555	1.4444	0.1111	1..222 2	...	0.0018
0004 Client	0.2857	0.6785	0.6428	1	0.9285	0.0714	0.7857	...	0.0011
0005 Client	0.3076	0.7307	0.6923	1.0769	1	0.0769	0.8461	...	0.0012
0006 Client	4	9.5	9	14	13	1	11	...	0.0164
0007 Client	0.3636	0.8636	0.8181	1.2727	1.1818	0.0909	1	...	0.0014
...	...	...	...	...	...	...	...	...	...
3000 Client	0.1777	0.4222	0.4	0.622	0.5777	0.044	0.4888	...	...

Table 5 Part of criteria table C3 - Source: Criteria table comparing 3000 clients for C3

C3	0001 Client	0002 Client	0003 Client	0004 Client	0005 Client	0006 Client	0007 Client	Normalized	Av
0001 Client	1	0.88666	0.9	0.9133	0.94	0.9533	0.9666	...	0.0017
0002 Client	1.1278	1	1.0150	1.0300	1.0601	1.0751	1.0902	...	0.0020
0003 Client	1.1111	0.9851	1	1.0148	1.0444	1.0592	1.0740	...	0.0019
0004 Client	1.0948	0.9708	0.9854	1	1.0291	1.0437	1.0583	...	0.0019
0005 Client	1.0638	0.9432	0.9574	0.9716	1	1.0141	1.0283	...	0.0019
0006 Client	1.0489	0.9300	0.9440	0.9580	0.9860	1	1.0139	...	0.0018
0007 Client	1.0344	0.9172	0.9310	0.9448	0.9724	0.9862	1	...	0.0018
...	...	...	...	...	...	...	...	...	...
3000 Client	1.0273	0.9109	0.9246	0.9383	0.9657	0.9794	0.9931	...	...

Table 6 Part of criteria table C4 - Source: Criteria table comparing 3000 clients for C4

C4	0001 Client	0002 Client	0003 Client	0004 Client	0005 Client	0006 Client	0007 Client	Normalized	Av
0001 Client	1	2.2949	1.2877	0.5683	1.3015	0.4964	1.3597	...	0.0010
0002 Client	0.4357	1	0.5611	0.2476	0.5673	0.2163	0.5924	...	0.0004
0003 Client	0.7765	1.7821	1	0.4413	1.0111	0.3854	1.0558	...	0.0008
0004 Client	1.7594	4.0379	2.2658	1	2.2911	0.8734	2.3924	...	0.0018
0005 Client	0.7679	1.7624	0.9889	0.4364	1	0.3812	1.0441	...	0.0008
0006 Client	2.0144	4.6231	2.5942	1.1449	2.6231	1	2.7391	...	0.0020
0007 Client	0.7354	1.6878	0.9470	0.4179	0.9576	0.3650	1	...	0.0007
...	...	...	...	...	...	...	...	...	...
3000 Client	0.6915	1.5870	0.8905	0.3930	0.9004	0.3432	0.9402	...	...

Finally, according to the aggregated vector of each table, the AHP systems reports, Table 7, the final list with the ranking of the potential clients who must be contacted firstly to save time and to indicate to the sales agent which are the most important to be called.

Table 7 Final ranking - Source: Unstructured database with one million transactions

Client	C1-Trip	C2-Road	C3-Card	C4-Country	Total
0946Client	0.001823687	0.004102874	0.00172917	0.028961351	0.011405662
0907Client	0.000632708	0.010940998	0.001822638	0.018100844	0.011381321
2672Client	0.003100267	0.016411497	0.001886367	0.007621408	0.011004299
0100Client	0.002384821	0.000841615	0.001983459	0.028961351	0.009841819
0939Client	0.002583556	0.000698362	0.001847606	0.028961351	0.009764345
0933Client	0.000968834	0.016411497	0.001954714	0.002732203	0.009364038
0006Client	0.002583556	0.016411497	0.001886367	0.002098649	0.009253103
0059Client	0.007750668	0.016411497	0.001886367	0.000837033	0.009165046
2923Client	0.002818425	0.016411497	0.00212402	0.00087233	0.008915695
0029Client	0.00081586	0.008205748	0.00168594	0.014480676	0.008879462
0109Client	0.003875334	0.016411497	0.002229343	0.000458249	0.008862725
0981Client	0.001291778	0.016411497	0.002075004	0.00098508	0.008854352
2038Client	0.001823687	0.016411497	0.00221107	0.000754202	0.008831455
2040Client	0.000861185	0.016411497	0.001913124	0.001026998	0.008821012
0020Client	0.003444741	0.016411497	0.001998152	0.000459704	0.008807793
2946Client	0.001069058	0.016411497	0.002059164	0.000827467	0.008790045
2878Client	0.00083791	0.008205748	0.002175407	0.01206723	0.008192265
...	...	...	...	...	...
from matrix C	0.059030356	0.499784984	0.12962052	0.31156414	

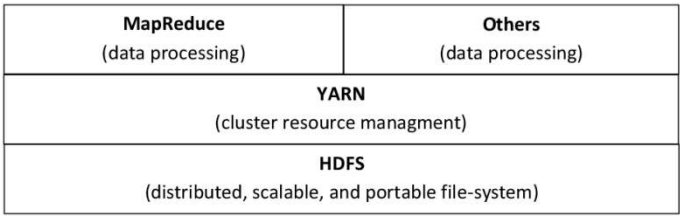

Figure 1 Hadoop ecosystem - own elaboration based on Apache Hadoop

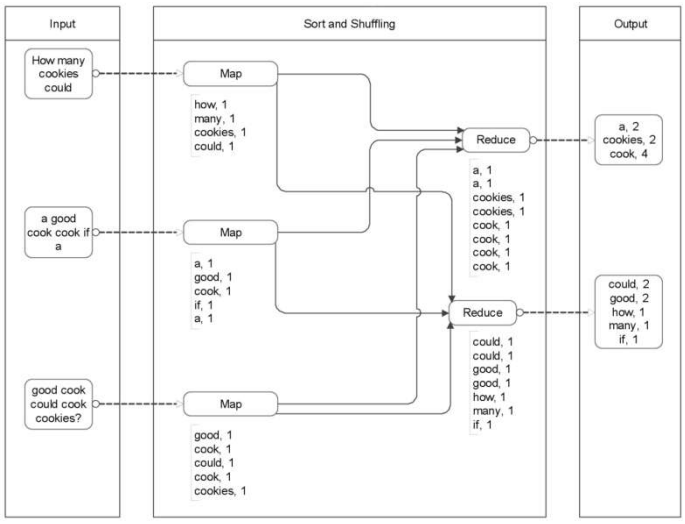

Figure 2 Simple MapReduce count task - own elaboration

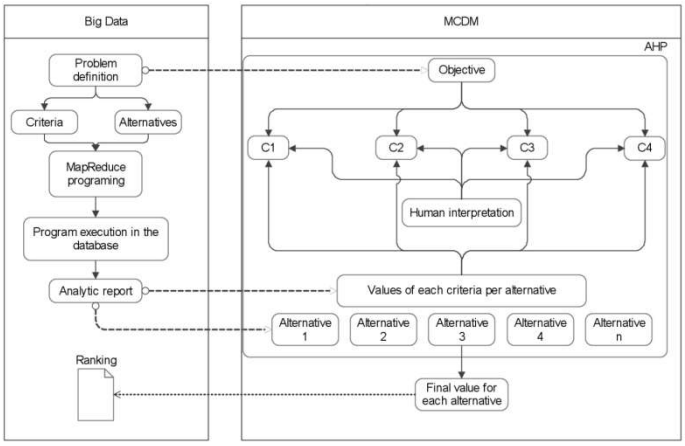

Figure 3 Working methodology

DISCUSSION

In this research, two well-known applications have been combined. On the one hand, the Analytic Hierarchy Process (AHP), which is one of the most used MCDM methodologies and on the other hand, big data analysis as one of the technologies which is growing faster to process large amounts of information in recent years. This combination allows us to extend the limits of AHP and other MCDM methodologies because the number of alternatives is not conditioned to human performance limitations. In other words, a huge number of alternatives can be processed as a human would but in few seconds.

When applying AHP methodology, the so-called 'Saaty scale' is used to carry out the pairwise comparison of the criteria for each alternative. Normally, this is done according to the knowledge and experience of a decision maker or based on the results of a previous survey conducted among a group of experts. This involves a high level of subjectivity, which is one of the characteristics of the AHP method. Without losing its fundamental subjectivity character but improving accuracy, the previous big data analysis allows us to make the pairwise comparisons not based on the knowledge of a single decision maker or a survey, but by using current and relevant data based on the alternative's behavior. The important question at this point is that, if a human could process such a large number of alternatives, the result of the study would be the same.

Due to the overwhelming quantities of data generated every day and since the cost of the storage systems is negligible nowadays, data are stored not only because there is a specific aim to use them, but also because they may be useful in the future. In this sense, traditional databases are not enough to store and process such huge amounts of data which are increased every day. Any person or machine connected to the Internet is a data generator and the trend is that it will be increased in the future. Thus, capable systems to process unstructured information and to report valuable data in short periods are needed.

Moreover, processing information properly allows us to relate data in a more effective way in order to know clients' habits and then, the possibility to get information in few seconds that would be cumbersome to obtain from surveys or relational databases. Knowing the clients' habits, companies are able to offer products in a more effective way while optimizing their resources. Among those resources, time is one of the most important ones and using big data tools as a previous step to the application of multicriteria decision making methods helps us to optimize the working time of the people involved in carrying out commercial tasks.

In addition, thanks to the case study described in Section 4, it has been demonstrated how MCDM methods can help to cover the gap of improving the predictive analysis and converting it into prescriptive analytics in which the added value is much higher

The presented methodology involves a Python code to adapt the basis of the MapReduce paradigm to a specific real case problem involving four criteria. As future working line, we propose to add one more criterion: frequency. By studying how often purchases of the same type of product have been made,

the accuracy, in terms of how frequent traveler a client is, will be increased.

References

Agrawal, D., El Abbadi, A., Das, S., and Elmore, A.J., Database scalability, elasticity, and autonomy in the cloud, Springer, (2011)

Andzulis, James "Mick" and Panagopoulos, Nikolaos G and Rapp, Adam, A review of social media and implications for the sales process, Journal *of Personal Selling & Sales Management*, Volume 32, Number 3, Pages 305--3016, Taylor & Francis, (2012)

Atzori, L., Iera, A. and Morabito, G., The internet of things: A survey, Computer networks, (2010)

Baby, M. *et al*., Comparative analysis of Cloud database, remote database, and traditional databasee, *International Journal of Computer Science and Information Security*, (2016)

Beaver, D., Kumar, S., Li H.C., Sobel, J., and Vajgel, P., Finding a Needle in Haystack: Facebook's Photo Storage, OSDI, (2010)

Beyer, M.A. and Laney, D., The importance of big data: a definition, Stamford, CT: Gartner, (2012)

Chen, H., Chiang, R., and Storey, V., Business intelligence and analytics: From big data to big impact, MIS quarterly, (2012)

Contini, B. and Zionts, S., Restricted bargaining for organizations with multiple objectives, Econometrica: *Journal of the Econometric Society*, (1968)

Cukier, K. and Mayer-Schoenberger, V., The rise of big data: How it's changing the way we think about the world, Foreign Aff., (2013)

Diebold, F., On the Origin (s) and Development of the Term Big Data, Penn Institute for Economic Research, (2012)

Gartner, I., Gartner it glossary devops, Gartner IT Glossary, (2017)

George, D., Mallery, P., IBM SPSS statistics 23 step by step: A simple guide and reference, Routledge, (2016)

Han, J., Haihong, E., Le, G., and Du, J., Survey on NoSQL database, IEEE, (2011)

Hays, C.L., What Wal-Mart knows about clients habits, The New York Times, (2004)

John Walker, S., Big data: A revolution that will transform how we live, work, and think, Taylor & Francis, (2014)

Kant, G., Jacks, M. and Aantjes, C., Coca-cola enterprises optimizes vehicle routes for efficient product delivery, Interfaces, (2008)

Kwong, C-K. and Bai, H., A fuzzy AHP approach to the determination of importance weights of Client requirements in quality function deployment, *Journal of intelligent manufacturing*, (2002)

LaValle *et al*., Big data, analytics and the path from insights to value, MIT sloan manage- ment review, Volume 52, Pages 21 (2011)

Maggino, F. and Ruviglioni, E., Obtaining weights: from objective to subjective approaches in view of more participative methods in the construction of composite indicators, Proceedings NTTS: New Techniques and Technologies for Statistics, (2009)

Mahadevan, B., Sivakumar, S., Dinesh Kumar, D. and Ganeshram, K., Redesigning midday meal logistics for the Akshaya Patra Foundation: OR at work in feeding hungry school children, Interfaces, (2013)

Marshall, Greg W and Moncrief, William C and Rudd, John M and Lee, Nick, Revolution in sales: The impact of social media and related technology on the selling environment, *Journal of Personal Selling & Sales Management*, Volume 32, Number 3, Pages 349--363, Taylor & Francis, (2012)

Menon, A., Big data@ facebook, Proceedings of the 2012 workshop on Management of big data systems, Pages 31-32, ACM, (2014)

Norusis, M.J., IBM SPSS statistics 19 guide to data analysis, Prentice Hall Upper Saddle River, New Jersey, (2011)

Ramanathan, Ramakrishnan and Mathirajan, Muthu and Ravindran, A Ravi, Big Data Analytics Using Multiple Criteria Decision-making Models, CRC Press, (2017)

Saaty, T.L., A scaling method for priorities in hierarchical structures, Journal of mathematical psychology, (1977)

Sapna, J., Comparative Study of Traditional Database and Cloud Computing Database, *International Journal of Advanced Research in Computer Science*, (2017)

Singh, Jagdip and Flaherty, Karen and Sohi, Ravipreet S and Deeter-Schmelz, Dawn and Habel, Johannes and Le Meunier-FitzHugh, Kenneth and Malshe, Avinash and Mullins, Ryan and Onyemah, Vincent, Sales profession and professionals in the age of digitization and artificial intelligence technologies: concepts, priorities, and questions, *Journal of Personal Selling & Sales Management*, Pages 1--21, Taylor & Francis, (2019)

Simon, H.A., Theories of bounded rationality, Decision and organization, (1972)

Underhill, P., Why we buy: The science of shopping– updated and revised for the Internet, the global consumer, and beyond, Simon and Schuster, (2009)

Xidonas, Panagiotis and Psarras, J., Equity portfolio management within the MCDM frame: a literature review, Vol 1, number 3, pp. 285--309, Inderscience Publishers, (2009)

Zionts, S. and Wallenius, J., An interactive programming method for solving the multiple criteria problem, Management science, (1976)

How to cite this article:

Mayor-Vitoria F *et al*., Big Data Analysis Tools Combined with AHP for Improving Bank Services Sales. *Int J Recent Sci Res.* 10(12), pp. 36216-36224. DOI: http://dx.doi.org/10.24327/ijrsr.2019.1012.4882

****