ANALYSIS OF STARTUP FUNDING IN INDIA

GROUP MEMBERS:

DARSHAN SHARMA

SUMANPREETI PHALGU

MRIDUL KAPRI , CEMK, REG NO- 1510701100 OF 2015-2016

SPANDAN MAITY, CEMK, REG NO- 151070110050 OF 2015-2016

DEBANJAN DAS, CEMK, REG NO- 151070110021 OF 2015-2016

CONTENTS

ACKNOWLEDGEMENT

I take this opportunity to express my profound gratitude and deep regards to my faculty **Mr. Titas Roy Chowdhury** for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him/her time to time shall carry me a long way in the journey of life on which I am about to embark.

I am obliged to my project team members for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

COLUMN NAME DESCRIPTION:

Columns	Types	Variable
SNo	Object	Continuous
Date	Object	Continuous
StartupName	Object	String Type Data
IndustryVertical	Object	Categorical
SubVertical	object	Categorical
CityLocation	Object	Categorical
InvestorsName	Object	String Type Data
InvestmentType	Object	Categorical
AmountInUSD	Object	Continuous
Remarks	Object	String Type Data

- There are 2371 rows and 10 columns in the Given Data.
- The record of Startup Funding from 02.01.2015 to 02.08.2017 in India.

TABLE CONTENTS:

I. **SNo.** : Serial no of the entries.
   - No missing value is here.

II. **Date** : The date of the data entry.
   - No missing value is here.

III. **StartupName** : The name of the startup company.
   - No missing value is here.

IV. **IndustryVertical** : Vertical Name of the company. In which category the company belongs to.
   - "consumer internet" is repeated 772 times.
   - "Technology" is repeated 313 times.
   - Ecommerce is repeated 230 times.
   - 171 values are missing. Total values are 2372. So percentage is : 7.2%.
   - And many values are repeated less number than that.

V. **SubVertical :** It is the sub division of the company category.
   - 936 values are missing. Total values are 2372. So percentage is : 39.46%.
   - "food delivery platform" is repeated 9 times.
   - "online lending platform" and "online pharmacy" is repeated 9 times.
   - And many values are repeated less number than that.

VI. **CityLocation** : The city name, from where the company has risen.
   - "bangalore" is repeated for 628 times.
   - "mumbai" is repeated for 446 times.
   - "new delhi" is repeated for 381 times.

- "gurgaon" is repeated for 240 times.
- 179 values are missing. Total values are 2372. So percentage is : 7.54%.

VII. **InvestorsName :** The name of the investors invested on the company.
- 8 values are missing. Total values are 2372. So percentage is : 0.34%.

VIII. **InvestmentType** : The type of investment done. Private Equity or Seed Funding.
- "seed funding" is repeated 1301 times.
- "private equity" is repeated 1068 times.
- Only 1 value is missing. Total values are 2372. So percentage is: 0.04%.

IX. **AmountInUSD** : The amount, that has been invested in the company.
- Here the missing values are not actually missing values, the startup companies have not actually got any investment.

X. **Remarks**: Comments given in some cases.
- Comments are optional. So, no missing value is here.

How we took care of null values:

We have converted the file into strings. And there we replaced all the strings where two commas are there one by one (",,") with the commas with a zero inside(",0,").

We could use "Null" instead of zero(0) . But there is a reason behind that.

On string type data we can consider "0" as string. And in the amount column we can consider "0" as numeric data type.

DATA PROCESSING

METHODS USED:

MAP REDUCE:

MapReduce is a processing technique and a program model for distributed computing based on java. The MapReduce algorithm contains two important tasks, namely Map and Reduce. Map takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs). Secondly, reduce task, which takes the output from a map as an input and combines those data tuples into a smaller set of tuples. As the sequence of the name MapReduce implies, the reduce task is always performed after the map job.

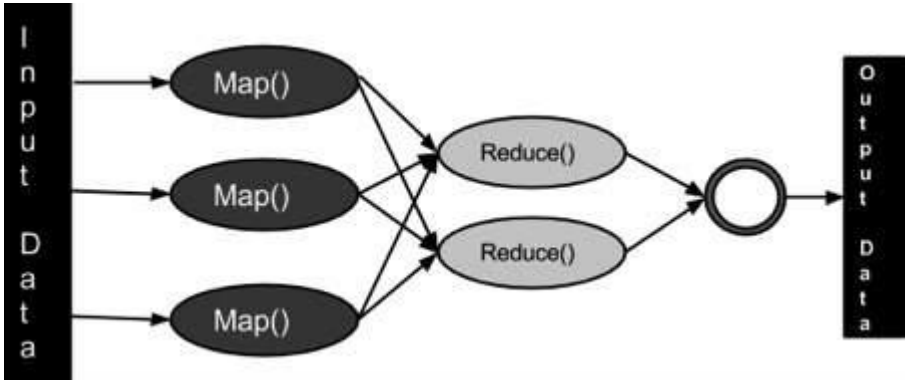
The major advantage of MapReduce is that it is easy to scale data processing over multiple computing nodes. Under the MapReduce model, the data processing primitives are called mappers and reducers. Decomposing a data processing application into mappers and reducers is sometimes nontrivial. But, once we write an application in the MapReduce form, scaling the application to run over hundreds, thousands, or even tens of thousands of machines in a cluster is merely a configuration change. This simple scalability is what has attracted many programmers to use the MapReduce model.

ALGORITHM OF MAPREDUCE:

- Generally MapReduce paradigm is based on sending the computer to where the data resides.
- MapReduce program executes in three stages, namely map stage, shuffle stage, and reduce stage.

- **Map stage** : The map or mapper's job is to process the input data. Generally the input data is in the form of file or directory and is stored in the Hadoop file system (HDFS). The input file is passed to the mapper function line by line. The mapper processes the data and creates several small chunks of data.
- **Reduce stage** : This stage is the combination of the Shufflestage and the Reduce stage. The Reducer's job is to process the data that comes from the mapper. After processing, it produces a new set of output, which will be stored in the HDFS.

- During a MapReduce job, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster.
- The framework manages all the details of data-passing such as issuing tasks, verifying task completion, and copying data around the cluster between the nodes.
- Most of the computing takes place on nodes with data on local disks that reduces the network traffic.
- After completion of the given tasks, the cluster collects and reduces the data to form an appropriate result, and sends it back to the Hadoop server.



INPUTS AND OUTPUTS(Java Perspective):

The MapReduce framework operates on <key, value> pairs, that is, the framework views the input to the job as a set of <key, value> pairs and produces a set of <key, value> pairs as the output of the job, conceivably of different types.

The key and the value classes should be in serialized manner by the framework and hence, need to implement the Writable interface. Additionally, the key classes have to implement the Writable-Comparable interface to facilitate sorting by the framework. Input and Output types of a MapReduce job: (Input) <k1, v1> -> map -> <k2, v2>-> reduce -> <k3, v3>(Output).

	Input	**Output**
Map	<k1, v1>	list (<k2, v2>)
Reduce	<k2, list(v2)>	list (<k3, v3>)

Example:

Word Count:

Input **(output of Map function)**	Set of Tuples	(Bus,1), (Car,1), (bus,1), (car,1), (train,1), (car,1), (bus,1), (car,1), (train,1), (bus,1), (TRAIN,1),(BUS,1), (buS,1), (caR,1), (CAR,1), (car,1), (BUS,1), (TRAIN,1)
Output	Converts into smaller set of tuples	(BUS,7), (CAR,7), (TRAIN,4)

Work Flow of Program:

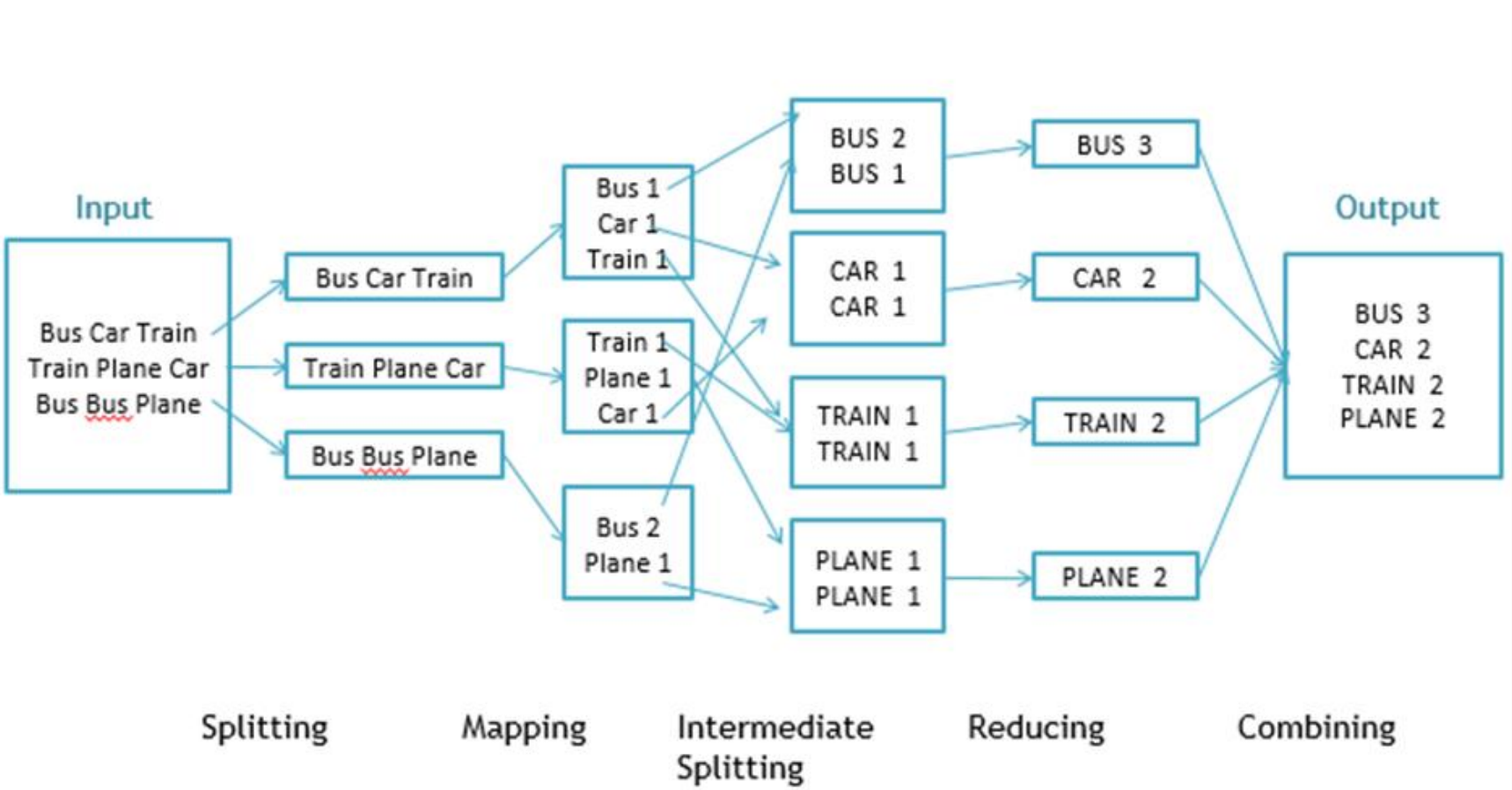


Fig. WorkFlow of MapReducing

STEPS OF WORKFLOW:

- **Splitting** : The splitting parameter can be anything, e.g. splitting by space, comma, semicolon, or even by a new line ('\n').
- **Mapping** : It takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (Key-Value pair).
- **Intermediate splitting** : the entire process in parallel on different clusters. In order to group them in "Reduce Phase" the similar KEY data should be on same cluster.
- **Reduce** :Takes the output from Map as an input and combines those data tuples into a smaller set of tuples.
- **Combining** : The last phase where all the data (individual result set from each cluster) is combine together to form a Result.

HIVE:

The Apache Hive (TM) data warehouse software facilitates reading, writing, and managing large datasets residing in distributed storage using SQL. Built on top of Apache Hadoop (TM), it provides:

- Tools to enable easy access to data via SQL, thus enabling data warehousing tasks such as extract/transform/load (ETL), reporting, and data analysis
- A mechanism to impose structure on a variety of data formats
- Access to files stored either directly in Apache HDFS (TM) or in other data storage systems such as Apache HBase (TM)
- Query execution using Apache Hadoop MapReduce, Apache Tez or Apache Spark frameworks.

Hive provides standard SQL functionality, including many of the later 2003 and 2011 features for analytics. These include OLAP functions, subqueries, common table expressions, and more. Hive's SQL can also be extended with user code via user defined functions (UDFs), user defined aggregates (UDAFs), and user defined table functions (UDTFs).

Hive users have a choice of 3 runtimes when executing SQL queries. Users can choose between Apache Hadoop MapReduce, Apache Tez or Apache Spark frameworks as their execution backend. MapReduce is a mature framework that is proven at large scales. However, MapReduce is a purely batch framework, and queries using it may experience higher latencies (tens of seconds), even over small datasets. Apache Tez is designed for interactive query, and has substantially reduced overheads versus MapReduce. Apache Spark is a cluster computing framework that's built outside of MapReduce, but on top of HDFS, with a notion of composable and transformable distributed collection of items called Resilient

Distributed Dataset (RDD) which allows processing and analysis without traditional intermediate stages that MapReduce introduces.

Users are free to switch back and forth between these frameworks at any time. In each case, Hive is best suited for use cases where the amount of data processed is large enough to require a distributed system.

Hive is not designed for online transaction processing. It is best used for traditional data warehousing tasks. Hive is designed to maximize scalability (scale out with more machines added dynamically to the Hadoop cluster), performance, extensibility, fault-tolerance, and loose-coupling with its input formats.

CODE AND QUERY

Program to Manipulate the given data:

Program Name: StrManipFinalEdit.java

```
import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Paths;

public class StrManipFinalEdit {
        public static void main(String[] args) throws Exception {
                String csvFile = "/home/edureka/startup_funding.csv";
                String content = new String(Files.readAllBytes(Paths.get(csvFile)));
              char c[]=content.toCharArray();
            boolean found=false;
            for(int i=0;i<c.length;i++){
                if (c[i]=='"') {
```

```
        found=!found;
    }
    if (c[i]==',') {
        if(found) {
            if(Character.isDigit(c[i-1])) {
                c[i]='_';
                }
            if(Character.isLetter(c[i-1])){
                c[i]='#';
            }
        }
    }
    if ((c[i]==',') && (c[i-1]==')')){
        if(found) {
        c[i]='#';
        }
    }
```

```
        if ((c[i]==',') && (c[i-1]=='.')){
                if(found) {
                c[i]='#';
                }
        }
        if (c[i]==';'){
                if(found) {
                c[i]='#';
                }
        }
        if ((c[i]==',') && (c[i-1]==' ')){
                if(found) {
                c[i]='#';
                }
        }
}
```

```
String str2 = String.valueOf(c);
String c2=str2.replaceAll("\"", "");
String c3=c2.replaceAll("_", "");
String c4=c3.replaceAll(",,", ",0,");
String c5=c4.replaceAll(",,", ",0,");
String c6=c5.toLowerCase();
String c7=c6.substring(c6.indexOf('\n')+1);

//#########################################################################
#######

BufferedWriter writer = null;
try
{
   writer = new BufferedWriter( new
FileWriter("/home/edureka/Startup_Funding_Modified.csv"));
   writer.write(c7);
```

```
            }
            catch ( IOException e)
            {
            }
            finally
            {
               try
               {
                  if ( writer != null)
                  writer.close( );
               }
               catch ( IOException e)
               {
               }
            }
         }
}
```

QUERIES FOR ANALYSIS:

1. **How does the funding ecosystem change with time?**
2. **Do cities play a major role?**
3. **Which industries are favored by investors for funding?**
4. **Who are the important investors in the Indian Ecosystem?**
5. **How much funds does startups generally get in India?**

ENTERING DATA IN HIVE:

```
hive>create table startup(Sno varchar(20),date varchar(20),companyName varchar(20),category varchar(40),subcategory varchar(50),location varchar(50),Iname varchar(150),Itype varchar(20),Amount float,remarks varchar(30)) row format delimited fields terminated by ',';
```

```
hive>load data local inpath '/home/edureka/Startup_Funding_Modified.csv' overwrite into table startup;
```

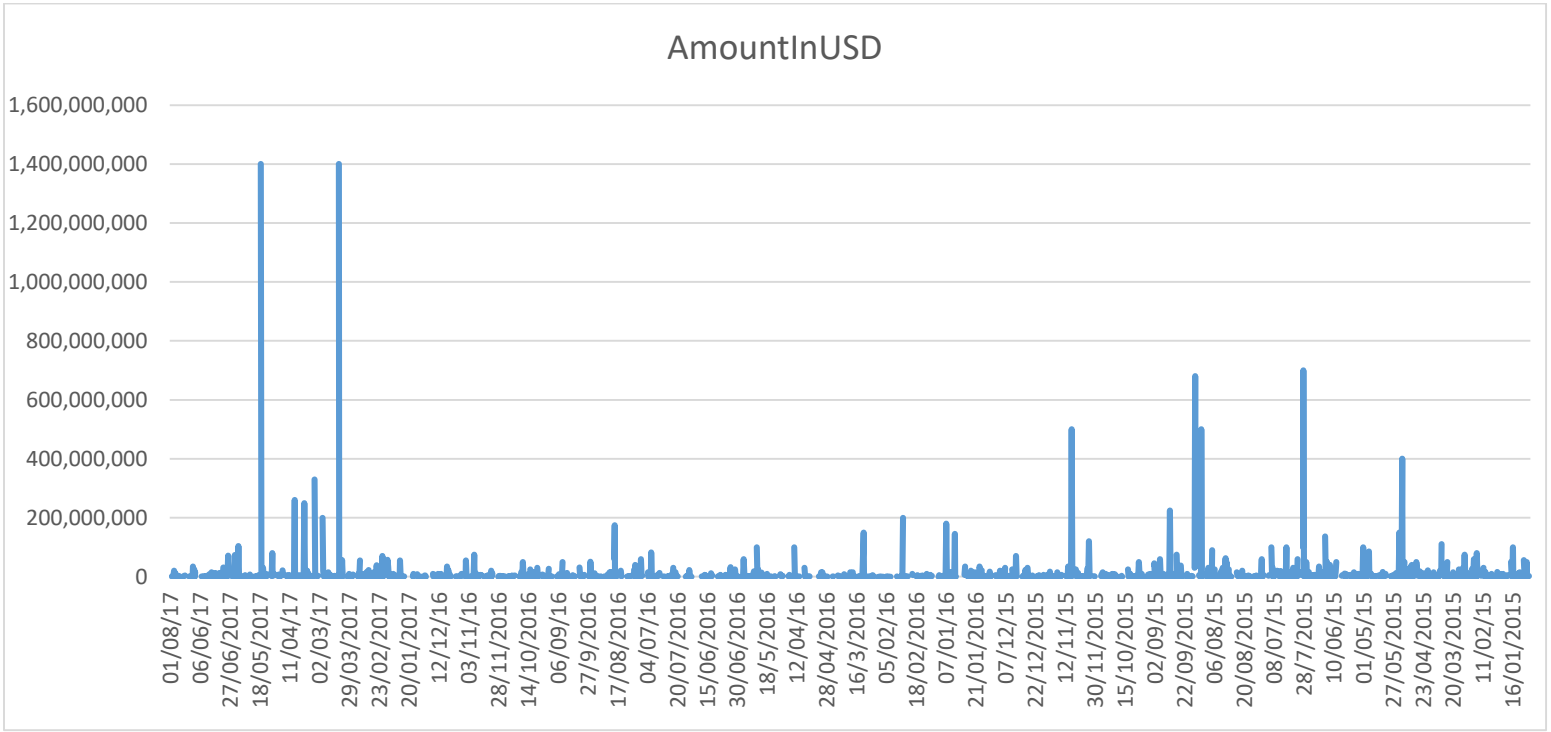
1. How does the funding ecosystem change with time?

Command:

hive> create table result1 as select date,sum(Amount) as SumAmount from startup group by date order by date asc;

hive>select * from result1;

Output:



Comment:

We can see that Investments in 2017 is higher than comparing to 2015. And in the year 2016 the investments are quite low.

2. Do cities play a major role?

Command:

```
hive>create table result2 as select sum(Amount) as SumAmount,location from startup group by location order by SumAmount desc;

hive>select * from result2;
```

Output:

```
8.383774108E9  bangalore
2.7502475E9    new delhi
2.3436945E9    mumbai
2.0678215E9    gurgaon
1.271863868E9  0
4.11105E8 chennai
2.82153E8 pune
1.94762E8 hyderabad
```

1.70338E8 noida

9.8186E7 ahmedabad

6.85E7 pune / us

6.7E7 new delhi / us

3.556E7 jaipur

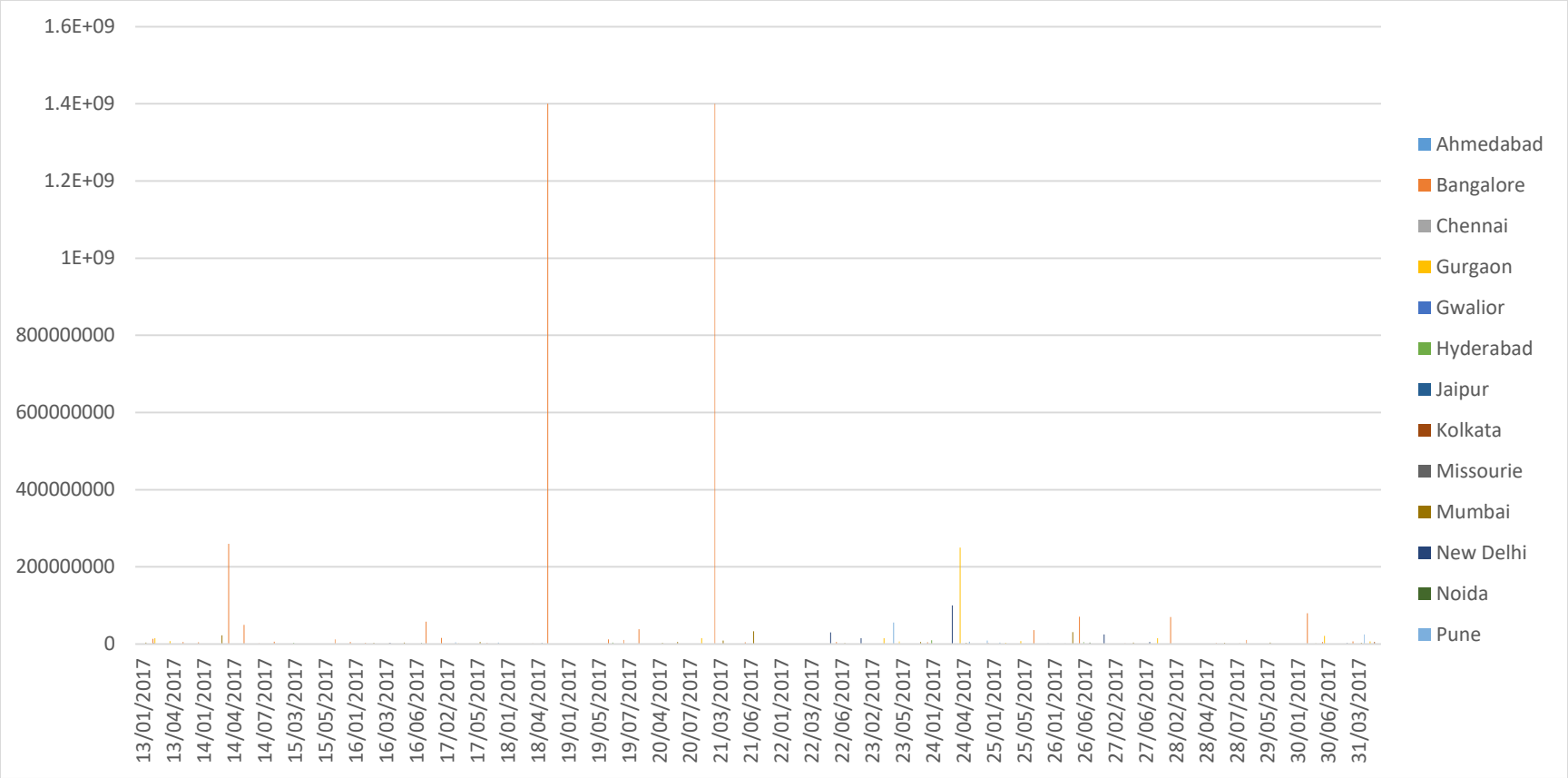
3.0E7 india / us

2.61E7 chandigarh

……..

….

….



Comment:

So the Bangalore based companies have a major investment in India.

Hence, it is proven that cities does play a role in investments.

3. Which industries are favored by investors for funding?

Command:

hive> create table result3 as select sum(Amount) as SumAmount,category from startup group by category order by SumAmount asc;

hive>select * from result3;

Output:

……..
……..
……..
2.25E8 cab rental mobile app
4.0E8 cab aggregator
5.0E8 ecommerce marketplace
5.0E8 car aggregator & retail mobile app
6.8E8 e-commerce & m-commerce platform
7.0E8 online marketplace
1.1035935E9 technology
3.797089E9 consumer internet
4.281189608E9 ecommerce
1.231811368E9 0

Comment:

'E-commerce' and 'consumer internet' vertical has the highest funding. And then comes missing valued funding, and then 'technology'. So the three industries are favored by investors.

4. Who are the important investors in the Indian Ecosystem?

Command:

hive> create table result4 as select sum(Amount) as SumAmount,Iname from startup group by Iname order by SumAmount asc;

hive> select * from result4;

Output:

……

……

……

1.25E8 ta associates

1.45E8 temasek holdings# march capital# warburg pincus

1.5E8 abraaj group

1.5E8 tiger global# investment ab kinnevik# steadview capital

1.75E8 tencent holdings# foxconn technology group# tiger global# softbank group# bharti enterprises

1.8E8 ctrip.com international ltd

1.8885E8 sequoia capital

2.0E8 alibaba

2.0E8 ontario teachers' pension plan & others

2.12E8 warburg pincus

2.25E8 falcon edge capital# ny based hedge fund# tiger global# softbank

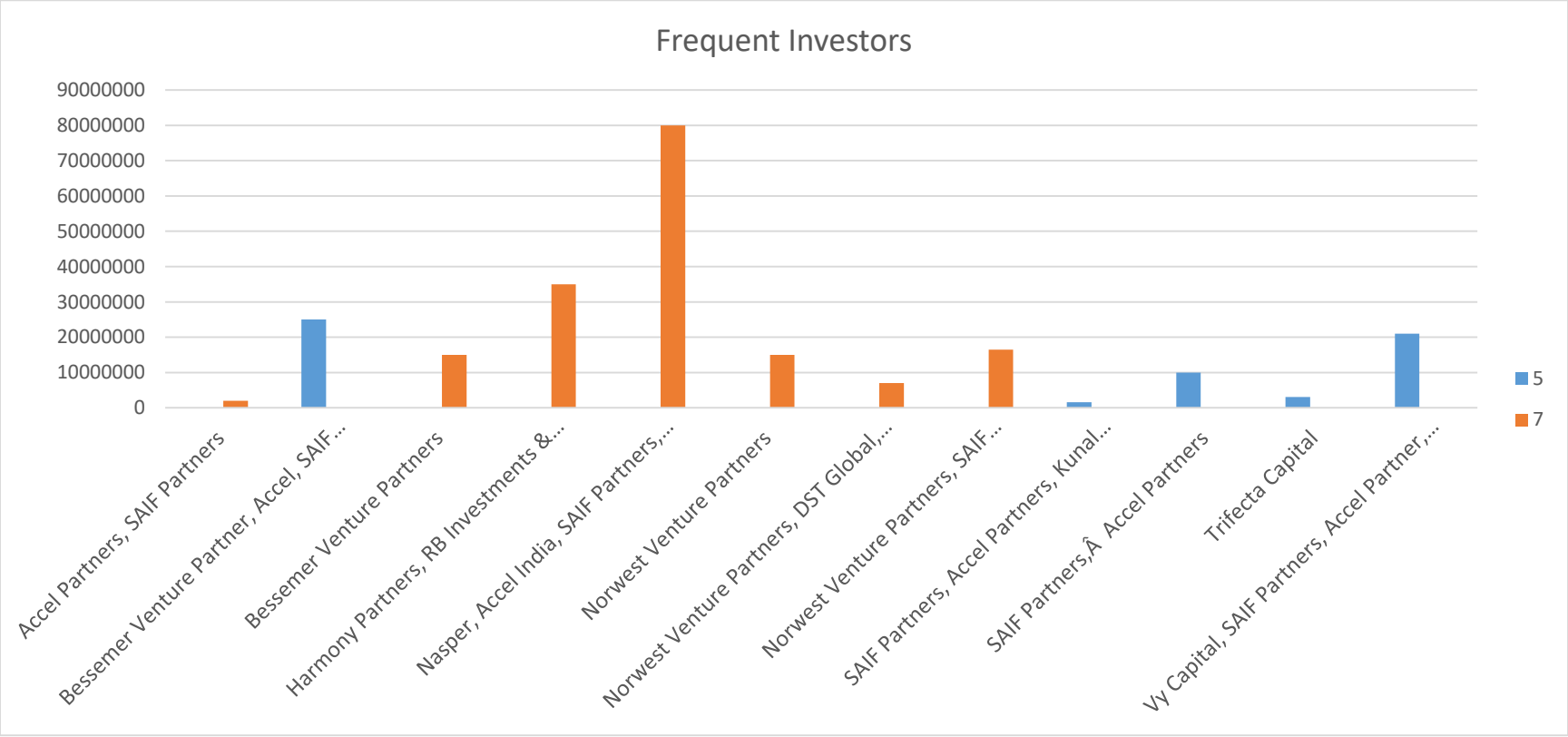
2.5E8 softbank vision fund# lightspeed venture partners# sequoia capital india advisors# greenoaks capital partners

2.6E8 simi pacific pte

3.3E8 softbank group corp

4.0E8 dst global# steadview capital# tiger global# accel partners & others

5.0E8	baillie gifford# falcon edge capital# tiger global# softbank group# dst global# didi kuaidi
5.0E8	alibaba# foxconn# softbank
6.8E8	alibaba group# ant financial
7.0E8	steadview capital and existing investors
1.4E9	microsoft# ebay# tencent holdings
1.467E9	softbank group



Comment:

Here we have counted the number of combined investments of several companies.

There are few companies that have invested 7 times and colored by Orange. And Few companies combined invested 5 times that have been colored by Green.

And here in the Orange category few companies like “Nasper”,”Accel India”,”SAIF Partners” etc. have done major investments combined.

5. How much funds does startups generally get in India?

Command:

```
hive>create table result5 as select avg(Amount) as SumAmount from startup where Amount<>'0';

hive>select * from result5;
```

Output:

1.2031073099016393E7

Comment:

The average funding in India is $ 1.2 X 10^7.

Command:

hive>create table result51 as select avg(Amount) as SumAmount,category from startup where Amount<>'0' group by category order by SumAmount;

hive>select * from result51;

Output:

……

……

.....

1.0E8	budget hotel accommodation
1.0E8	mobile advertising platform
1.2E8	hyper-local grocery delivery platform
1.37E8	logistics solution provider
1.5E8	online classifieds
2.25E8	cab rental mobile app
4.0E8	cab aggregator
5.0E8	car aggregator & retail mobile app
5.0E8	ecommerce marketplace
6.8E8	e-commerce & m-commerce platform
7.0E8	online marketplace

Comment:

The vertical wise average funding is not the same as it was on overall average that is 1.2×10^7. Here in 'online marketplace' vertical has the highest average. And then comes the

‘e-commerce & m-commerce platform’ and ‘ecommerce marketplace’ and so on.

Command:

hive>create table result52 as select avg(Amount) as SumAmount,location from startup where Amount<>'0' group by location order by SumAmount;

hive>select * from result52;

Output:

……..

……

…….

1.1E7 mumbai / global

1.2E7 udupi

1.2532251515151516E7 gurgaon

1.29728655660377 35E7 new delhi

1.5E7 pune/seattle

1.66E7 usa/india

1.7125E7 pune / us

2.07006768098765 43E7 bangalore

3.0E7	india / us
3.35E7	new delhi / us

Comment:

The city wise average funding is higher on that company which companies has a foreign company base. It is to be noticed that USA based companies has a great funding average.

CERTIFICATE

This is to certify that **Mr. DARSHAN SHARMA** has successfully completed a project on **"Big Data with Hadoop"** under the guidance of **Mr. Titas Roy Chowdhury.**

_____

[Mr. Titas Roy Chowdhury]

CERTIFICATE

This is to certify that **Mrs. SUMANPREETI PHALGU** has successfully completed a project on **"Big Data with Hadoop"** under the guidance of **Mr. Titas Roy Chowdhury.**

_____

[Mr. Titas Roy Chowdhury]

CERTIFICATE

This is to certify that **Mr. MRIDUL KAPRI** of **College of Engineering and Management,Kolaghat** , registration number: **151070110030 OF 2015-2016**, has successfully completed a project on **"Big Data with Hadoop"** under the guidance of **Mr. Titas Roy Chowdhury.**

_____

[Mr. Titas Roy Chowdhury]

CERTIFICATE

This is to certify that **Mr. DEBANJAN DAS** of **College of Engineering and Management,Kolaghat** , registration number: **151070110021 OF 2015-2016**, has successfully completed a project on **"Big Data with Hadoop"** under the guidance of **Mr. Titas Roy Chowdhury.**

_____

[Mr. Titas Roy Chowdhury]

CERTIFICATE

This is to certify that **Mr. SPANDAN MAITY** of **College of Engineering and Management,Kolaghat** , registration number: **151070110050 OF 2015-2016**, has successfully completed a project on **"Big Data with Hadoop"** under the guidance of **Mr. Titas Roy Chowdhury.**

_____

[Mr. Titas Roy Chowdhury]