

Data Mining Tools and Techniques

Social Profile Validation

Under Professor:
Dr. Kewei Sha

Team Members:

Nishil Prajapati
Sindhura Balagam
Silpa Jasthi

FINAL REPORT

Motivation:

It is highly impossible to determine a fake social profile for only one dataset. We can able to determine fraudulent of the profile by gazing at the multiple datasets which is a lot time consuming and demands of lot of effort to achieve. So by trying this, we will be able to learn more. If we do this an appropriate way, we can develop this project to a higher level.

Schedule Timings: Tuesday- 5pm to 10 pm

Sunday- 1pm to 5pm

Project Description:

Purpose of the Project:

Our purpose is to have three different datasets. We will analyze the profiles and decide whether it is fake profile or genuine. We want to determine how many fake and genuine profiles were found in large amount of records.. We want to achieve at least 30% - 50% probability for detecting genuine and fake profiles.

Expected outcome or result of the project:

Our main result is based on the user profile and the cluster in which that profile falls and also the neighbors surrounding the profile. This is calculated using the normalized values and distance values.

Evaluation Metrics:

- **Reliability:** To find whether the detected profiles are genuine or not.
- **Accuracy:** Comparing the accuracy of the techniques. To evaluate how effective the system works.

Plan of the Project:

1st week: Getting basic idea of the project.

2nd week: Search of research papers relevant to the project.

3rd week: Gathering the requirements needed for the project.

4th week: Defining the various dimensions for the datasets.

5th week: Datasets creation.

6th week: Overview of “R” language.

7th week: Study of Various techniques.

8th week: Getting the working environment setup –Rstudio.

9th week: Putting data into properly formatted text file.

10th week: Managing the multidimensional data using R.

11th week: Importing the datasets into R.

12th week: Design of the project.

13th week: Working and Implementation and testing outcomes and outliers.

Background research with bibliography of relevant research:

For the background research we searched for the journals and articles in IEEE. It gave an idea of the various techniques like similarity approaches, implementation and experiments that can be helpful in our project work. We learnt of the creation of datasets for our project. A group of records with basic fields are created by each member of the group (like Friends, Groups, Photos, Videos ...etc).

At presently we started coding the datasets in PHP language. So we are also having an overview of PHP language along with the R language. For Reference we have attached the link for IEEE paper here. We are still trying to find out some more papers on this topic. [1]

Detailed Research methodology or approach taken in the project:

A script file was created for the data fields. This script file is processed in the XAMPP software. Further a window pops up with all the fields(Friends, Groups, Videos, Photos) and later we inserted the values into it. The data which we added into the fields is seen in the MYSQL workbench through the SELECT command. The screenshots are displayed below.

Status of Implementation:

In this project we have used four technologies. We have created our own dataset in PHP and named it as “data.txt”. We have used two existing datasets collected from the youtube and web browser named as “User.txt” and “Users.tsv”.

Now the data is integrated by importing the existing and created datasets in R language. We used the following the function to import the three datasets.

```
dat <- read.table('data.txt')  
dat <- read.table('user.txt')
```

```
dat <- read.table('users.tsv')
```

We created the final data set by combining columns of different datasets. This is done by using the cbind function. Cbind function is used to combine the columns vertically.

```
dataset=cbind(data.frame(username[1]),data.frame(info[4]),data.frame(info[5]),data.frame(info[6]),data.frame(info[7]),data.frame(info[8]),data.frame(friend[1]),data.frame(photo[1]),data.frame(group[1]),data.frame(video[1]))
```

In order to minimize the redundant data we normalized the four values which are Friends, Groups, Videos, Photos between 0 and 1. This process of normalization is done in R language using normalize function(x). The following code shows how to normalize the four values.

```
normalize<- function(x)
{
x <- as.matrix(x)
minAttr=apply(x, 2, min)
maxAttr=apply(x, 2, max)
x <- sweep(x, 2, minAttr, FUN="-")
x=sweep(x, 2, maxAttr-minAttr, "/")
attr(x, 'normalized:min') = minAttr
attr(x, 'normalized:max') = maxAttr
return (x)
}
```

We expanded our data by finding the manhattan distance for each profile. This distance is calculated in R language using the function ManhattanD. So we represent all profiles using one value.

```
manhattanD=rowsums(new, na.rm=True)
```

Till Now, We just had data but not the label. So we had to decide the label. (Because We created our own dataset). We have decided the ranges for the profiles to be genuine according to Manhattan distance and individual normalized value. Now assigning label to profiles in PHP script.

Detection Techniques:

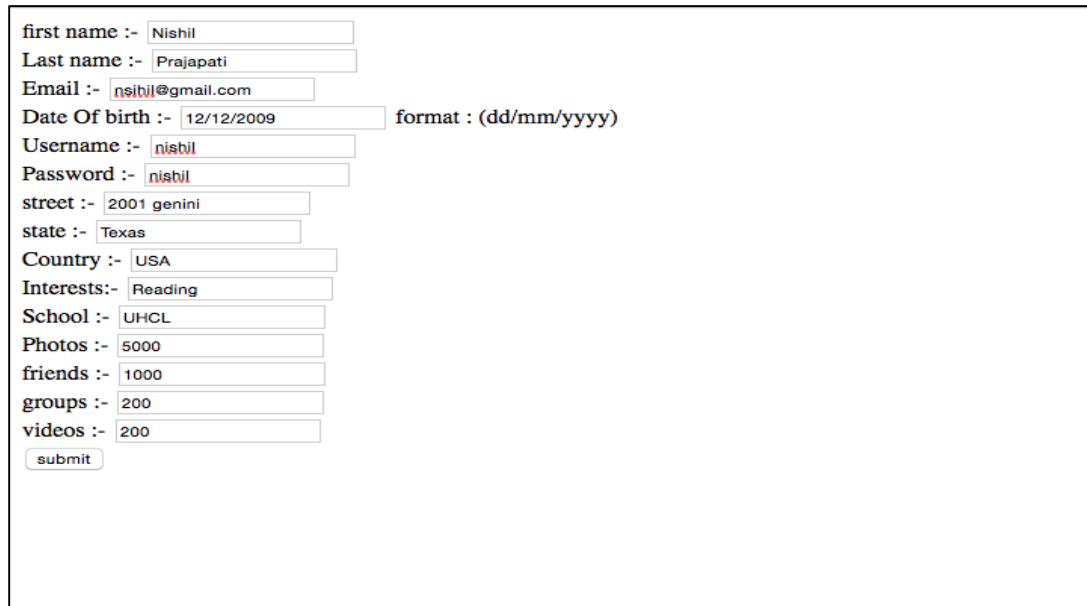
We used clustering technique based on Box plot. Box plot is a five-point summary. In this project we assumed that the input profile falls into three clusters. The cluster ranges from minimum to Q1 and the second cluster ranges from Q1 to Q3 and the third ranges from Q3 to maximum.

To find whether the profile is genuine or fake we will chose an input profile. Then find out the cluster and the neighbors based on Manhattan Distance and Normalized Values. We have

two types of labels 1 and 0 which are assigned to genuine and fake. Count the fake and genuine profiles nearest to your input profile and based on that decide your label. This implementation is done in Jsp , Core Java and MySQL to implement.

Evaluation Results:

After compiling the code and running the java scripts for the data sets the following window popup. Our result is based on the four fields which are photos, friends, groups and videos.



A screenshot of a web form used for profile evaluation. The form contains the following fields and values:

- first name :- Nishil
- Last name :- Prajapati
- Email :- nishil@gmail.com
- Date Of birth :- 12/12/2009 format : (dd/mm/yyyy)
- Username :- nishil
- Password :- nishil
- street :- 2001 genini
- state :- Texas
- Country :- USA
- Interests:- Reading
- School :- UHCL
- Photos :- 5000
- friends :- 1000
- groups :- 200
- videos :- 200

At the bottom of the form is a "submit" button.

Output:

According to Manhattan Distance, Profile Fake

According to Friends Normalized Value, Profile Genuine

According to Photos Normalized Value, Profile Fake

According to Groups Normalized Value, Profile Fake

According to Videos Normalized Value, Profile Genuine

Conclusion:

As we don't have our datasets, creation of datasets is the major task of our project. Later integrating the three datasets and finding the normalized values and distances for all the profiles is the second task. After using all the technologies like R, PHP, JAVASCRIPT, SQL we are able to get good results i.e., finding the social profile to be genuine or fake. We can say that our results met our expectations.

Future work:

In this datamining project we found the friends for the input profile. It would attain more good results if we consider the profile's circle. It would be exciting if we find the mutuality scenario between two profiles and their relation between them.

Program:

We had attached two codes PHP and java script. They are used to fetch the data from the database and in calculating the manhattan distance. Java code is used to find whether the profile is genuine or Fake.

PHP:

```
$username = "root";
$password = "";
$hostname = "localhost";
//connection to the database
$dbhandle = mysql_connect($hostname, $username, $password)
    or die("Unable to connect to MySQL");
echo "Connected to MySQL<br>";

$selectd = mysql_select_db("R",$dbhandle)
    or die("Could not select examples");

$result = mysql_query("SELECT manhattanD FROM Manhattan limit
300000");
$result1 = mysql_query("select V1,V2,V3,V4 from
Profile_Normalized");

//fetch the data from the database
while (($row = mysql_fetch_array($result)) && ($row1 =
mysql_fetch_array($result1))) {
    //echo $row1[0];
    echo $row[0];
    if($row{'manhattanD'} >= 0 && $row{'manhattanD'} <= 1)
    {
```

```

        $query = "
INSERT                                INTO                                `R`.`final`
(`Friend`,`photo`,`group`,`video`,`manhattanD`,`class`)      VALUES
('".$row1[0]."', '".$row1[1]."', '".$row1[2]."', '".$row1[3]."', '"
$row[0]."',0);"
mysql_query($query);
    }
    else if($row{'manhattanD'} > 1 && $row{'manhattanD'} <= 2)
    {
        $query = "
INSERT                                INTO                                `R`.`final`
(`Friend`,`photo`,`group`,`video`,`manhattanD`,`class`)      VALUES
('".$row1[0]."', '".$row1[1]."', '".$row1[2]."', '".$row1[3]."', '"
$row[0]."',1);"
mysql_query($query);
    }
    else if($row{'manhattanD'} > 2 && $row{'manhattanD'} <= 2.2)
    {
        $query = "
INSERT                                INTO                                `R`.`final`
(`Friend`,`photo`,`group`,`video`,`manhattanD`,`class`)      VALUES
('".$row1[0]."', '".$row1[1]."', '".$row1[2]."', '".$row1[3]."', '"
$row[0]."',0);"
mysql_query($query);
    }
    else if($row{'manhattanD'} > 2.2 && $row{'manhattanD'} <=
2.759)
    {
        $query = "
INSERT                                INTO                                `R`.`final`
(`Friend`,`photo`,`group`,`video`,`manhattanD`,`class`)      VALUES
('".$row1[0]."', '".$row1[1]."', '".$row1[2]."', '".$row1[3]."', '"
$row[0]."',1);"
mysql_query($query);
    }
    else if($row{'manhattanD'} > 2.759 && $row{'manhattanD'} <= 3)
    {
        $query = "
INSERT                                INTO                                `R`.`final`
(`Friend`,`photo`,`group`,`video`,`manhattanD`,`class`)      VALUES
('".$row1[0]."', '".$row1[1]."', '".$row1[2]."', '".$row1[3]."', '"
$row[0]."',0);"
mysql_query($query);
    }
    else if($row{'manhattanD'} > 3 && $row{'manhattanD'} <= 3.25)
    {
        $query = "

```

```

        INSERT                                INTO                                `R`.`final`
        (`Friend`,`photo`,`group`,`video`,`manhattanD`,`class`)      VALUES
        ('".$row1[0]."', '".$row1[1]."', '".$row1[2]."', '".$row1[3]."', '".
        $row[0]."',1);";
        mysql_query($query);
    }
    else
    {
        $query = "
        INSERT                                INTO                                `R`.`final`
        (`Friend`,`photo`,`group`,`video`,`manhattanD`,`class`)      VALUES
        ('".$row1[0]."', '".$row1[1]."', '".$row1[2]."', '".$row1[3]."', '".
        $row[0]."',0);";
        mysql_query($query);
    }
}
mysql_close($dbhandle);
?>

```

Java Program:

```

package data;
import java.sql.DriverManager;
import java.sql.Connection;
import java.sql.*;
import java.sql.SQLException;
/* *To change this license header, choose License Headers in
Project Properties.
 * To change this template file, choose Tools | Templates* and
open the template in the editor. */
/** * @author nishil09*/
//package current;
public class ProfileMatch
{
    public NewClass nnn(double dis,double f,double p,double
g,double v
    {
        System.out.print(dis);
        int[] sl= new int[300000];
        Connection con = null;
        String a = "",b = "",c = "",d = "",e = "";
        Try
        {
            Class.forName("com.mysql.jdbc.Driver");
            double limit1 = dis - 0.5;
            double limit2 = dis + 0.5;

```



```
con = null;
con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/R",
"root", "");
Statement st = con.createStatement();
String s = "Select * from final where manhattanD >= " + limit1
+" "+ "and" + " " + " manhattanD" + "<=" + " " + limit2;
// String s2 = "Select manhattanD from Manhattan where
manhattanD > " + 1.596 + " "+ "and" + " " + " manhattanD" + "<="
+" " + 2.406;
// String s3 = "Select manhattanD from Manhattan where
manhattanD > " + 2.406 + " "+ "and" + " " + " manhattanD" + "<="
+" " + 3.895;
ResultSet rs = st.executeQuery(s);
// ResultSet rs1 = st1.executeQuery(s2);
//ResultSet rs2 = st2.executeQuery(s3);
double size = 0.0;
int genuine = 0;
int fake = 0;
while(rs.next())
{
    // System.out.println((Integer)rs.getObject("class"));
    if(rs.getInt(5) == 1)
    {
        genuine++;
    }
    else
    {
        fake++;
    }
    // s1[counter] = rs.getInt("V1"); //
}
//con.close();
if(genuine > fake)
{
    a = "According to Manhattan Distance, Profile Genuine";
}
else
{
    a = "According to Manhattan Distance, Profile Fake";
}
genuine = 0;
fake = 0;
limit1 = f - 0.1;
limit2 = f + 0.1;
```

```
s = "Select * from final where Friend >= " + limit1 + " " + "and"
+ " " + " Friend" + "<=" + " " + limit2;
rs = st.executeQuery(s);
while(rs.next())
{
    //
    System.out.println((Integer)rs.getObject("class"));
    if(rs.getInt(5) == 1)
    {
        genuine++;
    }
    else
    {
        fake++;
    }
}
if(genuine > fake)
{
    b = "According to Friends Normalized Value, Profile
Genuine";
}
else
{
    b = "According to Friends Normalized Value, Profile
Fake";
}
genuine = 0;
fake = 0;
limit1 = p - 0.1;
limit2 = p + 0.1;
s = "Select * from final where photo >= " + limit1 + " " +
"and" + " " + " photo" + "<=" + " " + limit2;
rs = st.executeQuery(s);
while(rs.next())
{
    // System.out.println((Integer)rs.getObject("class"));
    if(rs.getInt(5) == 1)
    {
        genuine++;
    }
    else
    {
        fake++;
    }
}
if(genuine > fake)
{
```

```
        c = "According to Photos Normalized Value, Profile
Genuine";
    }
    else
    {
        c = "According to Photos Normalized Value, Profile
Fake";
    }
    genuine = 0;
    fake = 0;
    limit1 = g - 0.1;
    limit2 = g + 0.1;
    s = "Select * from final where final.group >= " + limit1 + "
"+ "and" + " " + " final.group" + "<=" + " " + limit2;
    rs = st.executeQuery(s);
    while(rs.next())
    {
        // System.out.println((Integer)rs.getObject("class"));
        if(rs.getInt(5) == 1)
        {
            genuine++;
        }
        else
        {
            fake++;
        }
    }
    if(genuine > fake)
    {
        d = "According to Groups Normalized Value, Profile
Genuine";
    }
    else
    {
        d = "According to Groups Normalized Value, Profile Fake";
    }
    genuine = 0;
    fake = 0;
    limit1 = v - 0.1;
    limit2 = v + 0.1;
    s = "Select * from final where video >= " + limit1 + " "+ "and"
+ " " + " video" + "<=" + " " + limit2;
    rs = st.executeQuery(s);
    while(rs.next())
    {
        // System.out.println((Integer)rs.getObject("class"));
```

```
        if(rs.getInt(5) == 1)
        {
            genuine++;
        }
        else
        {
            fake++;
        }
    }
    if(genuine > fake)
    {
        e = "According to Videos Normalized Value, Profile
Genuine";
    }
    else
    {
        e = "According to Videos Normalized Value, Profile
Fake";
    }
    // double result = (genuine / fake) * 100.0;
    NewClass obj = new NewClass(a,b,c,d,e);
    // System.out.print(result);
    // String res = "Profile is genuine by " + result + " % " +
"Gen" + genuine + " fake" + fake;
    return obj;
}
catch(ClassNotFoundException e1)
{
    System.out.print("Something Went wrong");
}
catch(SQLException e1)
{
    System.out.print("Something Went wrong");
e1.printStackTrace();
}
finally
{
}
return null;
}
}
```

Responsibilities of team members:

Being newbies in this topic, all the team members took part in all phases of project development. We acquire good knowledge of all the phases like creation of datasets, importing into R, merging the datasets, and designing the algorithm etc.

References:

[1] Raad, E.; Chbeir, R.; Dipanda, A., "User Profile Matching in Social Networks," in *Network-Based Information Systems (NBIS), 2010 13th International Conference*.