**MINOR PROJECT REPORT**



**Facebook event analyser**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Name | Enrollment | Batch | Contact no | Email Id |
| 1. | Aviral Goyal | 15103321 | B-8 | 9205498184 | aviral.goyal38@gmail.com |
| 2. | Sajal Subodh | 15103332 | B-8 | 8368459948 | sajal.subodh10297@gmail.com |
| 3. | Shivam Chimra | 15103337 | B-8 | 8377038833 | s.shivam2601@gmail.com |

**Submitted To *-* Dr. Adwitiya Sinha**

**Contents**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Topic** | **Page No.** |
| 1. | Abstract | 2 |
| 2. | Introduction | 3 |
| 3. | Related Work | 4 |
| 4. | Model Assumptions | 5-8 |
|  | 1. Data Source and Dataset | 5 |
|  | 2. Tools/Techniques and Libraries used | 6-7 |
|  | 3. Methods/ Algorithms used | 7-8 |
| 5. | Experimental Outcomes | 9-19 |
|  | 1. User Interface | 9 |
|  | 2. Analysis and Findings | 10-19 |
|  | 2.1 Pie Charts | 10-12 |
|  | 2.2 Geo Plots | 12-15 |
|  | 2.3 Wordcloud | 15-16 |
|  | 2.4 Machine Learning | 17-19 |
| 6. | Conclusion | 20 |
| 7. | References | 20 |
|  |  |  |

**Abstract**

In the recent time facebook has exponentially expanded its target audience, it's being used by diverse audience which adds on to the advantage of hosting various events to attract maximum audience. Though for hosting an event certain factors need to be kept in mind to reach maximum people to attend our event.Thus through this project we aimed to create a web application which can be used to analyse the most important features in predicting the count of people who will attend our event. We analysed around around 1000 events of 5 categories(travel, food, music, art and education) with the help of various statistical plots like pie chart, geo plot and word cloud, which demonstrate that which event should be organized where, who should be our target audience and how we need to market our event to attract maximum audience. We also used machine learning models to predict the interested count to our event. Our web application can be used by the organizers of the event to make better decision and attract more crowd.

**Introduction**

The enormous growth of facebook usage has led to accumulation of huge data which can be used to extract the present trends and patterns out of it. These pattern can be used for predictive analysis. We therefore carried out research on the facebook event data to build a predictive model, in which we found out the various important features used to analyse our data and made it fit for prediction.

Like we all know the importance of location and other factors like social media marketing while organizing an event as same event in location X with better content shared on social media will have more interested people than one with Y location with average content shared on social media.But how can organizers come to know about this.With the help of this project we intend to provide a solution to these organizers.We are going to collect events data from various sources like facebook and others.After preprocessing data or getting important details extracted from data we will make proper analysis of various factors playing role in attracting crowd and then will make a model trained on this dataset so that for future events we could predict expected crowd on basis of location,tickets price,post content on social media,etc.We will make web app where we can input an Event ID and our trained model predicts the interested\_count on it.The WebApp will have cool visualizations of the data that we have collected.

With help of this organizers would be able to make better decisions to attract more crowd.

The data we worked upon was collected from facebook through Graph API, fetching the data was a challenging task in itself as in R programming there isn’t a good support for graph API as compared to python.So we first extracted the event ID of all the events category wise and stored it in JSON format and then through these id we fetch the data of events. Also we were restricted to limited data as we can extract no more than 1000 data observation at a particular time, which conflict with the accuracy of our machine learning model. Integrating the whole project in a web application was a challenging task as we had to integrate each and every feature into our app and adding each feature into the web application includes a lot of code at front as well as at backend.

Though despite of the challenges we managed to create an application which stands out and can be used by event organizer to plan out a successful event. A event organizer can use our app to analyze the market trend and plan out events keeping in mind the interest of audience. We have used pie-chart to show which category event attract how much audience, so that organizers can focus more on that category and for the event having less audience they can rectify that where are they going wrong and how can they improve. We used geo-plotting to demonstrate the trend of various events in particular location so that a organizer can host its event to pull maximum crowd.Further we used the description of each event to extract the important word which can be used in the event description to attract more audience. Finally we applied two machine learning models to predict the interested count of a events based on the input features. The machine learning models we used are random forest and XGBoost, and we also plotted the feature importance plot to demonstrate the features most important in our predictive analysis.

**Related Work**

Facebook’s Graph API is not only limited to providing events data only,it can give access to many other personnel details of a user to another user.[Zubeida Casmod Khan](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin="Authors":.QT.Zubeida Casmod Khan.QT.&newsearch=true) and [Thulani Mashiane](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin="Authors":.QT.Thulani Mashiane.QT.&newsearch=true) have discussed this in detail in their paper ‘An analysis of Facebook's graph search’

They used two separate profiles and executed graph queries and analysed results as if they were positive or negative

"Crawling Facebook for Social Media Analysis Purposes" by Salvatore A. Catanese discusses in very good way about development of tools which helps in analyzing properties of social network graphs,describing connection between participants to social networks, it very well demonstrate how we can work on big datasets and use it for our predictive analysis. This paper very well demonstrated the use mongodb and how can it be used to store data by extracting from an API.

In order to prepare word cloud we will need relevance of each word so Juan Ramos has explained very nicely in his paper about using tf-idf method to calculate relevance.

After analysis when we will be implementing random forest method for prediction,we can use better subset of decision trees by removing decision trees with bad results.[Simon Bernard](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin="Authors":.QT.Simon Bernard.QT.&newsearch=true) has discussed this in great detail in his paper.

“XGBoost: A Scalable Tree Boosting System” by Tianqi Chen and Carlos Guestrin.Through this paper we got to know XGBoost is based on gradient boosting and got to know how boosting actually work.Chen very well demonstrated that XGBoost improves the basic algorithm into an advance algorithm for prediction.It can be used to solve problems related to regression as well as classification and ranking problems just by changing the parameters. In the paper Chen and Guestrin very well explained the various parameters of the algorithms and when should be a particular parameter be changed. The paper also described the various losses in the algorithm and which parameter can be used to train our model to minimize that loss.

**Model Assumptions**

**Data Source and Dataset**

The data of events from around the world was collected from Facebook using its Graph API belonging to following five categories :

I. Travel

II. Food

III. Music

IV. Art

V. Education

Approximately 250 events from each of these categories (total 1250 events) was fetched from the Graph Api and stored into a MongoDB database.

The dataset consists of 14 major attributes listed below:

1. Name *(String)* - The event name
2. Description *(String)* - The description posted by owner for the event
3. Start Time *(Time)* - Start time of the event
4. End Time *(Time) -* End Time of the event
5. Timezone *(String)* - The timezone of the event. This is of the form *Country-code/Region.*
6. Latitude (*Real) -* The latitude of the event
7. Longitude (*Real) -* The longitude of the event.
8. Type *(String) -* The category of the event *i.e travel, food, music, art or education*.
9. Attending\_count *(integer)* - count of people attending the event
10. Declined\_count *(integer)* - count of people who declined the event
11. Interested\_count *(integer)* - count of people who marked “Interested” in the event
12. Maybe\_count *(integer)* - same as interested count
13. Noreply\_count *(integer)* - count of no-reply of invitees in an event.
14. Updated\_time *(Time)* - The most recent time of the event (if updated by the owner).

**Tools/ Techniques and Libraries used**

1. First the event id for the category wise events was extracted and stored in a JSON file.
2. The stored ids were used to create API calls and make GET requests to Facebook and fetch the event data. This data was stored in a MongoDB database.

**Packages used** -Rfacebook, rjson, jsonlite, httr, RMongo, mongolite.

1. The web application for the analysis tool was created using **RShiny** library of R. Shiny is an R package that makes it easy to build interactive web apps straight from R. The applications consists of a user interface and the server side controls all user inputs and generates necessary outputs/plots.
2. **ui.R-** The R script is uses a R package ‘Shiny’ to provide UI to our application.

**Packages used**- shiny, plotly, shinythemes, ggplot2, maps.

B. **server.R-** The R script provides the backend to our application.This file

load data into the application based on the user input and is responsible for creating all kinds of plots.

**Packages used**- shiny, plotly, shinythemes, ggplot2, maps.

1. 3 different kind of plots for analysis are created- Donut charts, Geoplots and Word Clouds. All the scripts are present in the folder d3-charts.
   1. **d3-charts/piechart.R** : This script reads data from the database, separates the attributes like Attending Count, Interested Count, Maybe Count, No Reply Count , Declined Count and creates a dataset pie\_chart\_data.csv for creating useful pie charts for analysis and comparing values of these attributes across categories. **Packages used** - RMongo, Mongolite.
   2. **d3-charts/geochart.R :** The script is used to generate geo-map based on the event type.. This script creates by dataset geo\_chart\_data.csv by separating the latitude and longitude from the original dataset.

**Packages Used**- mongolite, ggplot2

* 1. **d3-charts/DataPreProcessing.R :** The script fetches data from MongoDB database and performs text pre-processing on the Description attribute of dataset. The NLP techniques are applied for preprocessing to compute word frequencies and create document-term matrix for each category and saves it into 5 five separate file namely dtm-travel.csv, dtm-food.csv, dtm-music.csv, dtm-art.csv and dtm-education.csv. The **tm** package is used for text preprocessing and text mining.

**Packages Used-** mongolite, tm, SnowballC

V. After analysis we applied machine learning algorithms for prediction of interested count in future based on features like Attending\_count, Maybe\_count, Can\_guest invite, Guest\_list\_enabled, No\_reply\_Count.

Two models namely Random Forest and XGBoost ere applied which resulted in approx 85% accuracy. Here accuracy measure is correlation coefficient.We also prepared plots to find importance of feature in prediction.

**Packages Used-** caTools,randomForest, xgboost

**Methods/ Algorithms**

**Natural Language Processing Techniques for Data Pre-processing and Data Cleaning**

Text cleaning was required on the attribute “Description” of the dataset.

We perform the following steps to make sure that the text mining we’re dealing with is clean for event descriptions belonging to each category :

* Eliminate extra white spaces, tabs and new line characters
* Removing non ASCII characters (some descriptions were in non-english language which were removed)
* Convert the text to lower case, so that words like “write” and “Write” are considered the same word for analysis
* Remove numbers
* Remove English stop words e.g “the”, “is”, “of”, etc
* Remove punctuation e.g “,”, “?”, etc
* Eliminate remaining white spaces
* Stemming our text
* Create a document-term matrix for that category and store it in a .csv file.

(**Packages used - tm, SnowballC**)

**Random Forest**

Random Forest is one of most easy to use machine learning algorithm without much use of hyperparameter tuning.This is an ensemble learning method in which multiple decision trees are trained on data sample with help of bootstraping.Final prediction is function of prediction from each model,it can be mean of predictions from each decision tree.

Another great quality of the random forest algorithm is that it is very easy to measure the relative importance of each feature on the prediction.Through looking at the feature importance, you can decide which features you may want to drop, because they don’t contribute enough or nothing to the prediction process.

Hyperparameters :

**N\_estimators :** No. of trees to be used.

**Max\_features :** maximum number of features Random Forest is allowed to try in an individual tree.

**min\_sample\_leaf** : This determines the number of leafs.

**n\_jobs** : how many processors it is allowed to use

**Oob\_score :** This is a random forest cross validation method. In this sampling, about one-third of the data is not used to train the model and can be used to evaluate its performance

**XGBoost**

XGBoost stands for **eXtreme Gradient Boosting.**

Three main forms of gradient boosting are supported:

* **Gradient Boosting algorithm** also called gradient boosting machine including the learning rate.
* **Stochastic Gradient Boosting** with sub-sampling at the row, column and column per split levels.
* **Regularized Gradient Boosting** with both L1 and L2 regularization.

The XGBoost library implements the **gradient boosting decision tree algorithm**.

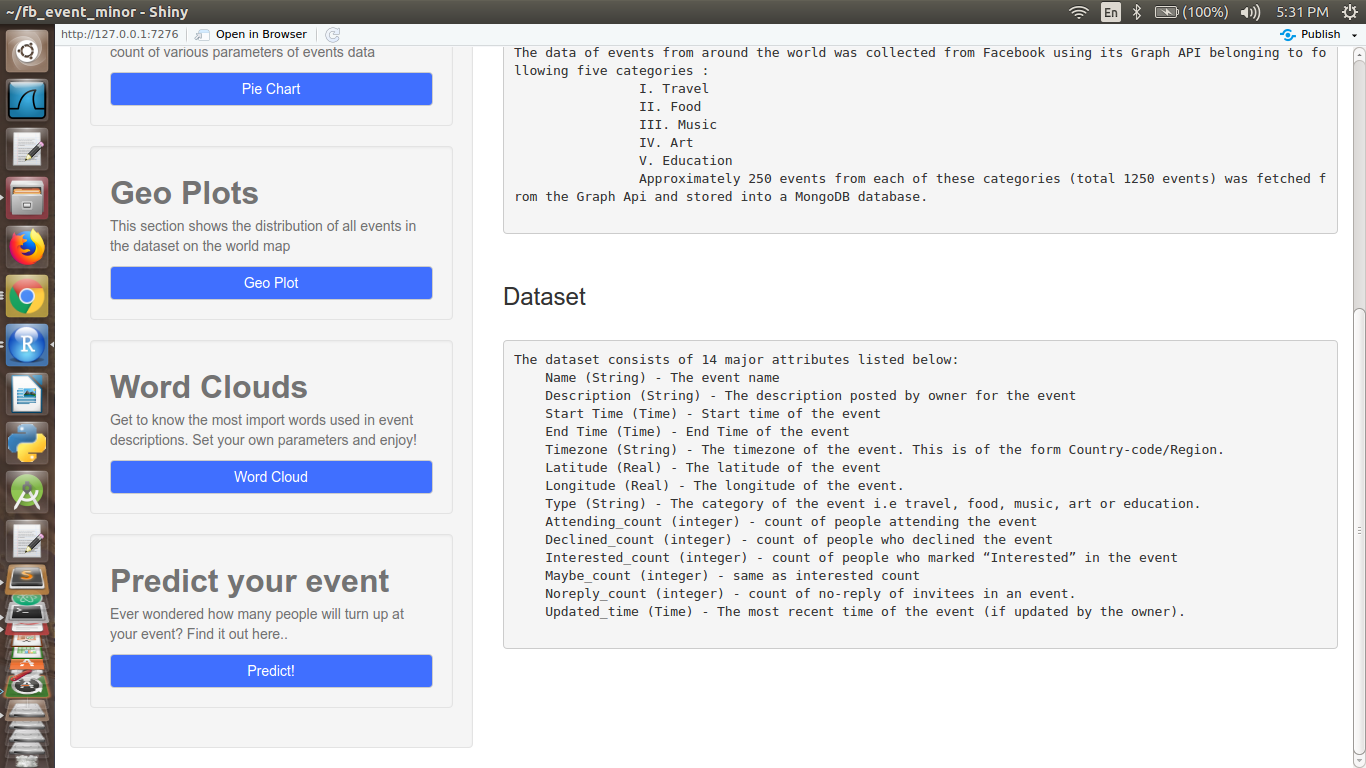
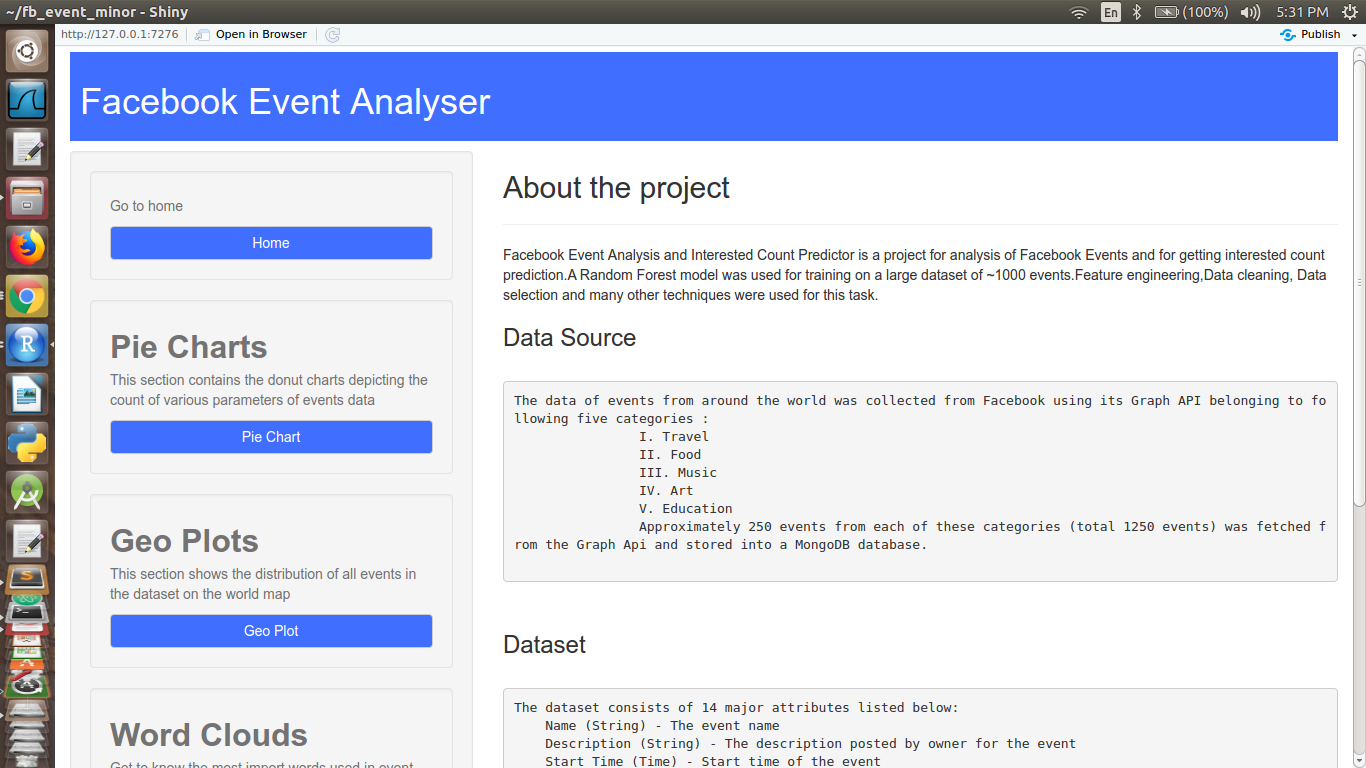
This algorithm goes by lots of different names such as gradient boosting, multiple additive regression trees, stochastic gradient boosting or gradient boosting machines.

Boosting is an ensemble technique where new models are added to correct the errors made by existing models. Models are added sequentially until no further improvements can be made. Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models and then added together to make the final prediction. It is called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models.

This approach supports both regression and classification predictive modeling problems.

**Experimental Outcomes**

The web application built on Shiny had a simple, understandable and user friendly user interface.



*Fig.1: The User Interface*

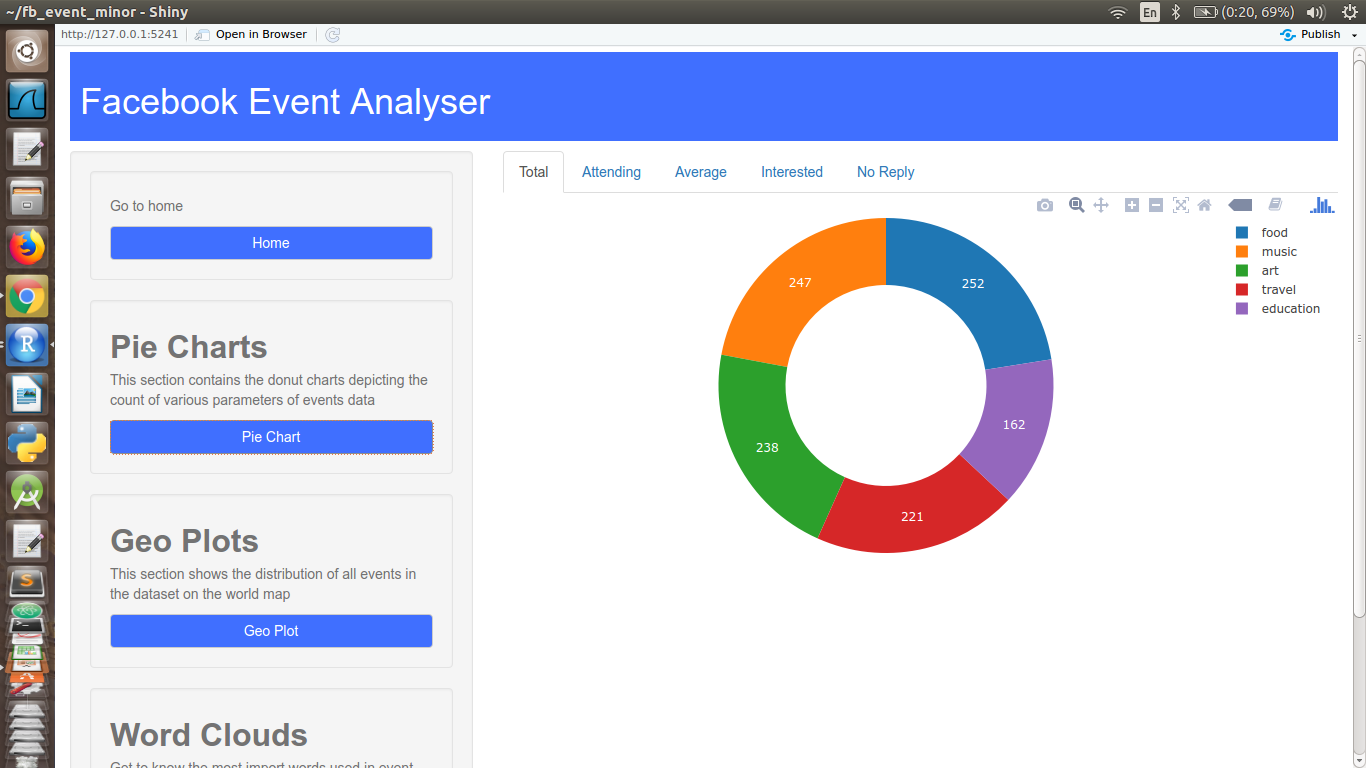
It consists of two panels : the **main panel** and a **side panel**. The side panel consists of following functionality buttons and the main panel is used to output based on the button selection.

1. Home - the introductory page giving brief introduction of the data source, dataset and the models used.
2. Pie Chart - to show the donut plots depicting the fraction of various attributes in the dataset.
3. Geo Plots - to show the distribution of events category-wise on a world map and region-wise count of events on a histogram.
4. Word Cloud - to show a word cloud of important and most-frequently used words in event description on a wordcloud.
5. Predict - to predict an event’s interested count.

**Analysis and Findings**

**Pie Charts**

1. **Total Count**



*Fig.2: Total event counts*

The plot gives the following results:

Food - 252

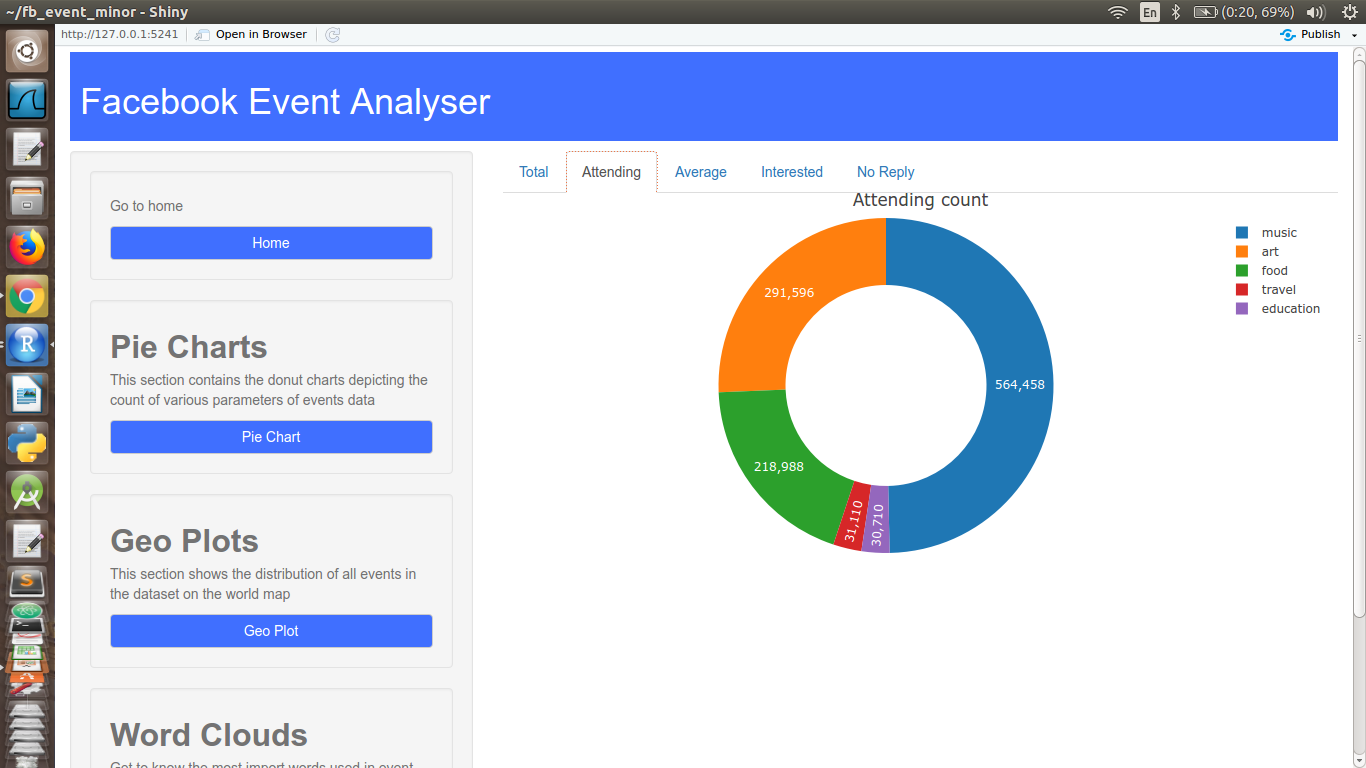
Music - 247

Art- 238

Travel- 221

Education- 162

1. **Attending Count**



*Fig.3: Total attending count*

The plot gives the following results:

Music - 564,458 (49.7%)

Art - 291,596 (25.6%)

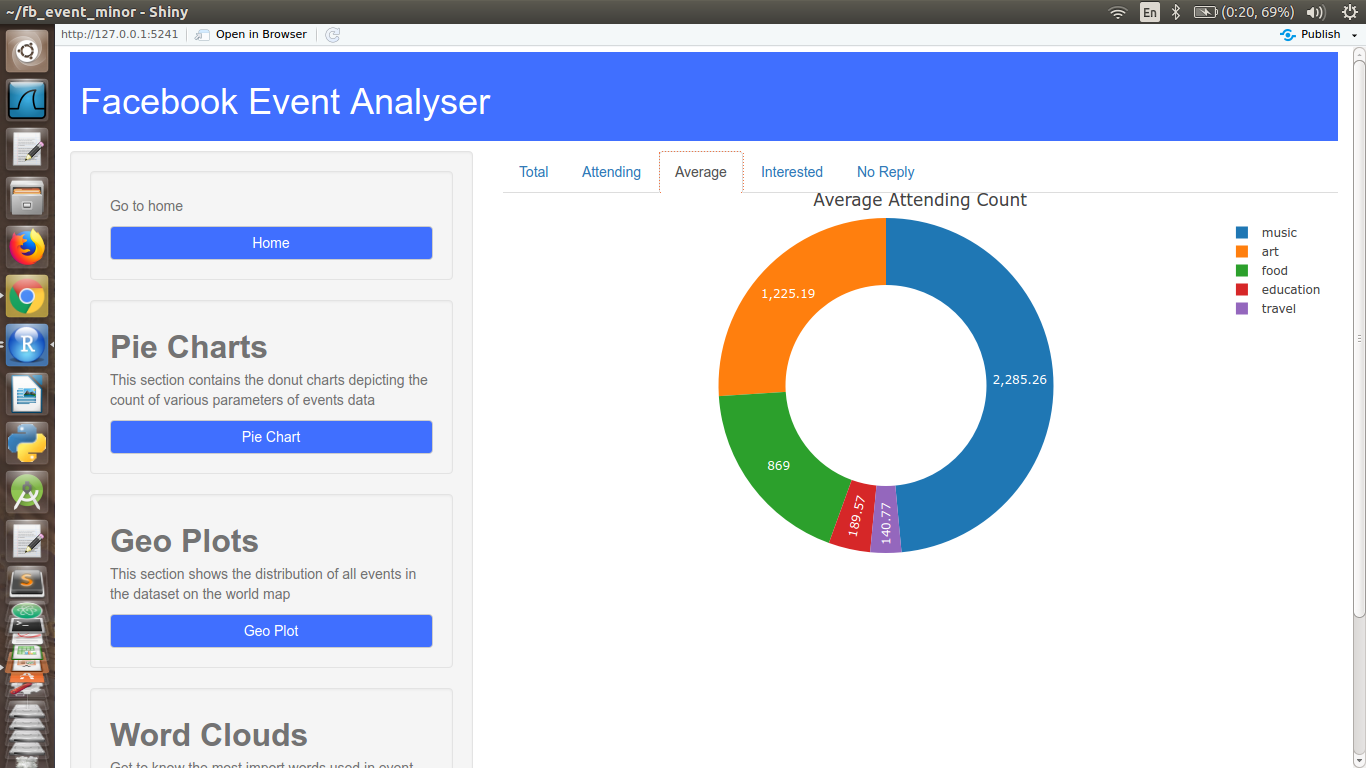
Food- 218,988 (19.3%)

Travel- 31,110 (2.74%)

Education- 30,710 (2.7%)

Clearly, the people going in a music related event are much more than any other category.

1. **Average Attending Count**



*Fig.4: Average attending count*

The plot gives the following results:

Music - 2,285.26 (48.5%)

Art - 1,225.19 (26%)

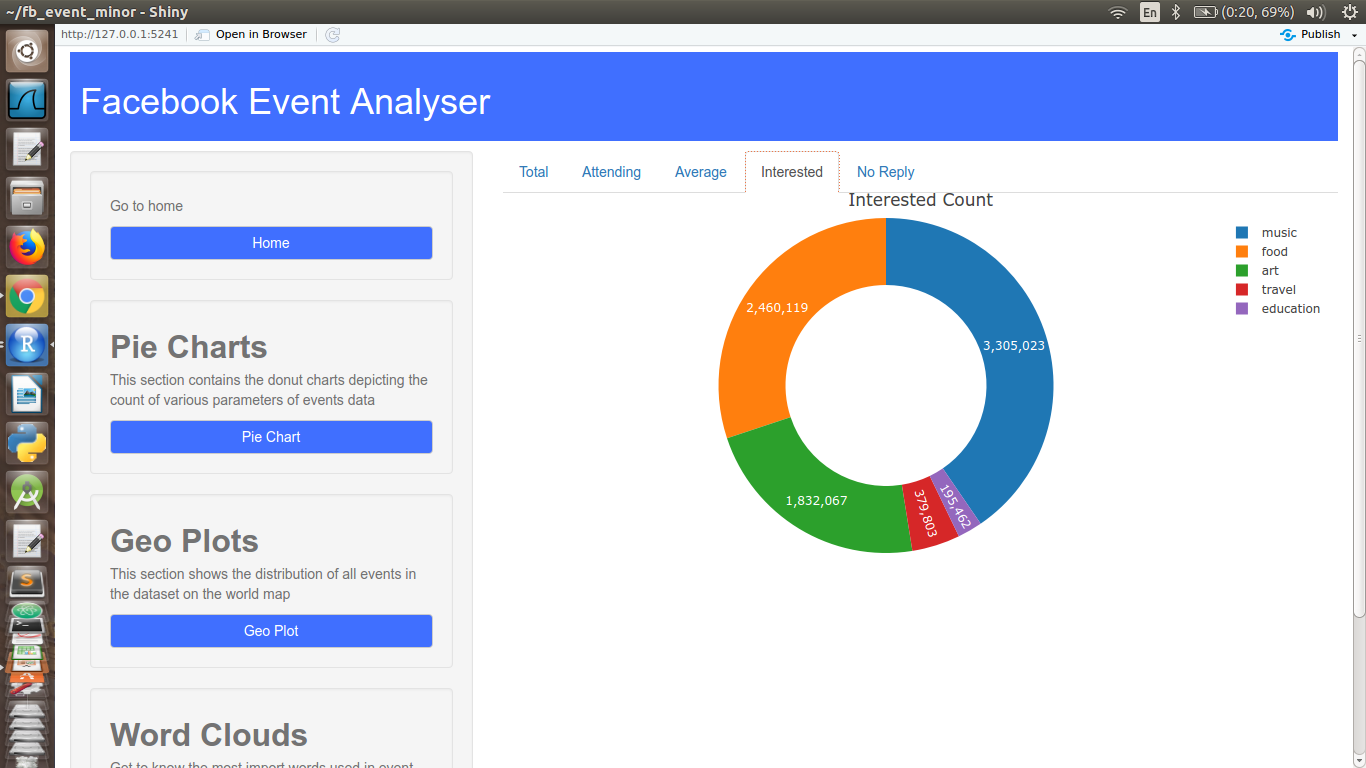
Food- 869 (18.5%)

Travel- 189.57 (4.03%)

Education- 140.77 (2.99%)

As expected, the average people going in a music related event are much more than any other category.

1. **Interested Count**



*Fig.5: Interested count*

The plot gives the following results:

Music - 3,305,023 (40.4%)

Art - 1,832,067 (22.4%)

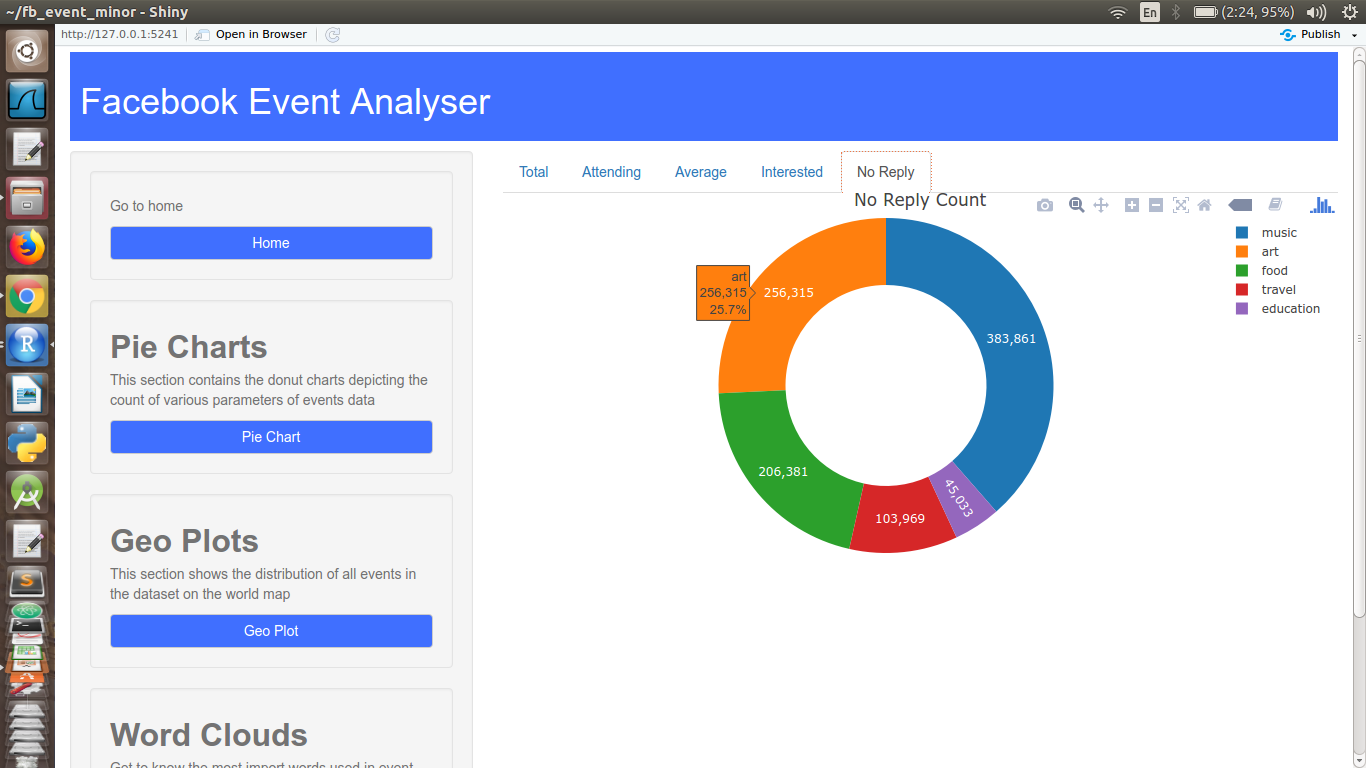
Food- 2,460,119 (30.1%)

Travel- 379,803 (4.64%)

Education- 195,462 (2.39%)

The interested count is still more in music, but people are more interested in food events than art events, still more people attend art events than food events.

1. **No reply Count**



*Fig.6: No reply count*

The plot gives the following results:

Music - 383,861 (38.6%)

Art - 256,315 (25.7%)

Food- 206,381 (20.7%)

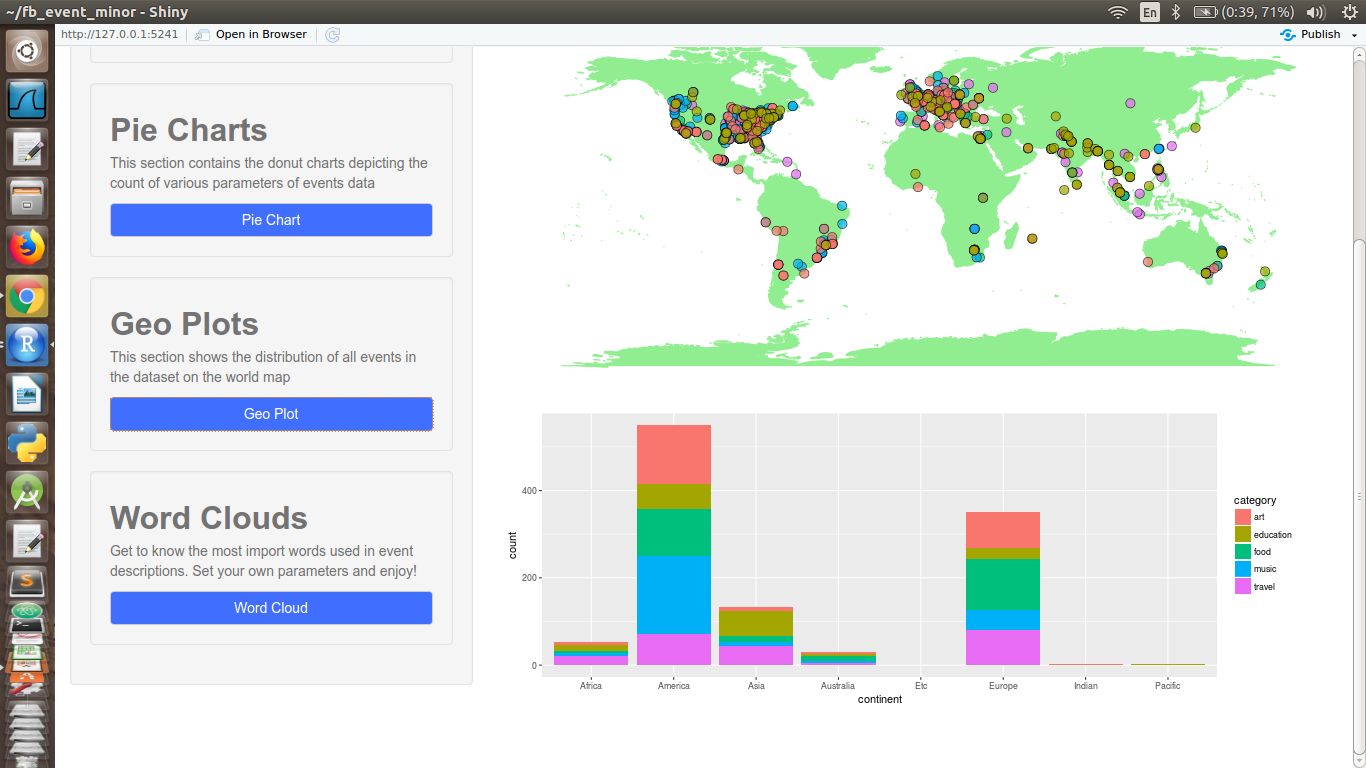
Travel- 103,969 (10.4%)

Education- 45,033 (4.52%)

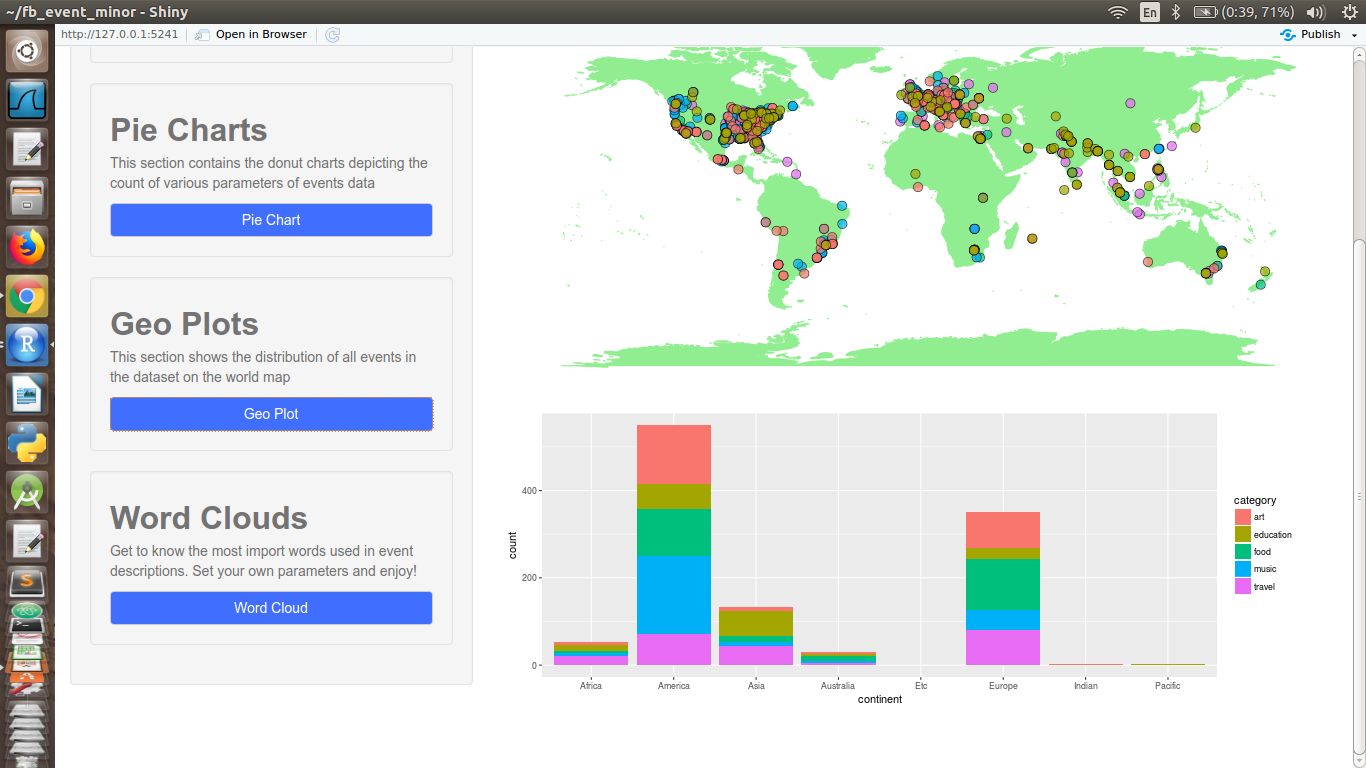
Thus, the people not replying to invites in music related event are much more than any other category.

**Geo-plots**

The distribution of events on the world map is as shown by the fig. 7 and fig 8. Shows the frequency distribution based on continents.



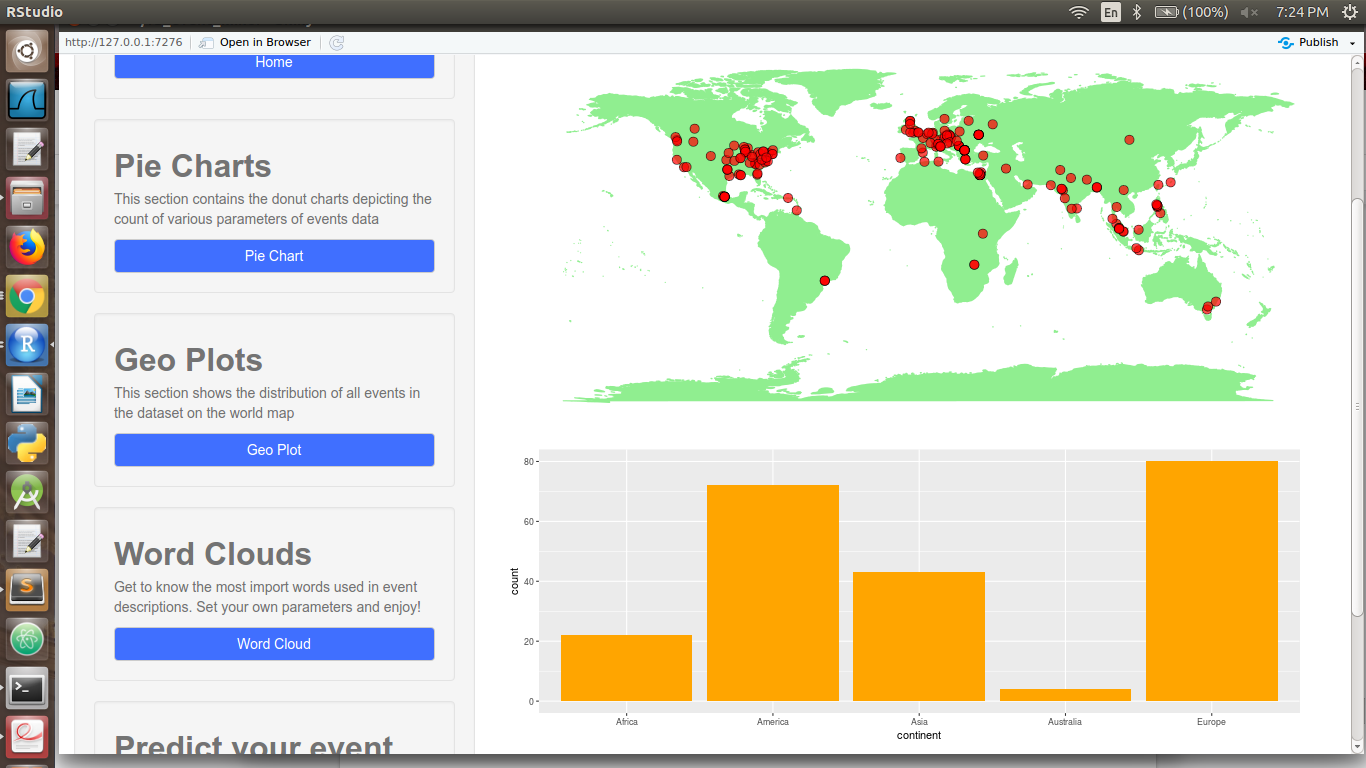
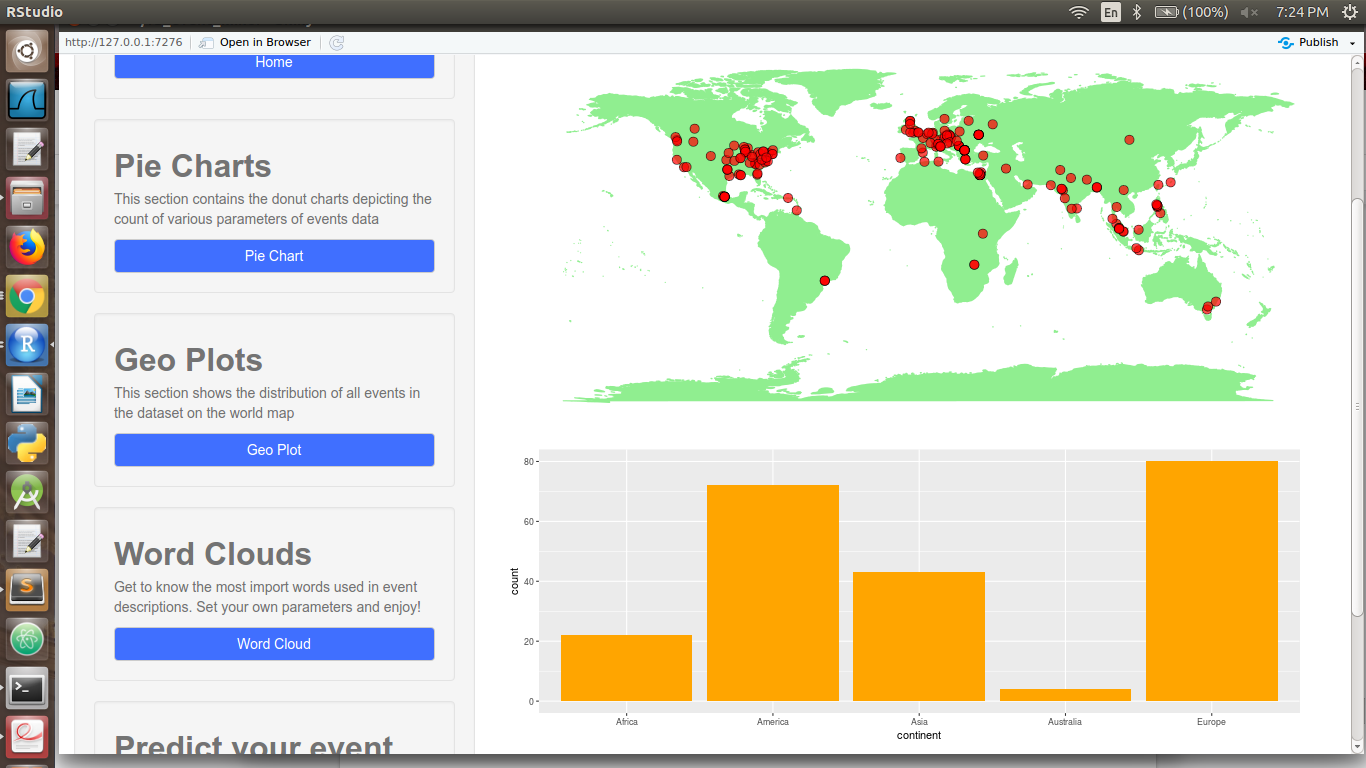
*Fig.7: Geo-distribution of all events*



*Fig.8: Frequency of events distributed by continents*

The histogram shows maximum events happening in America and Europe.

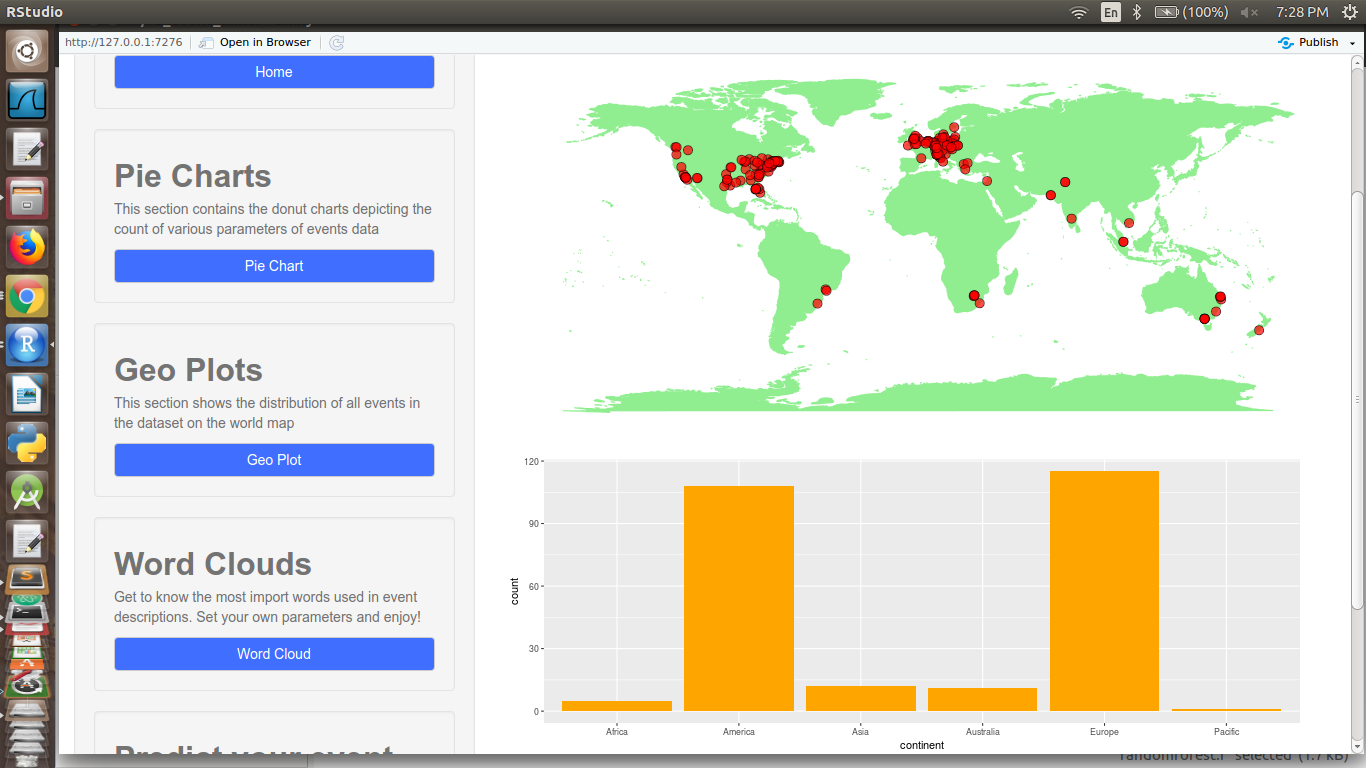
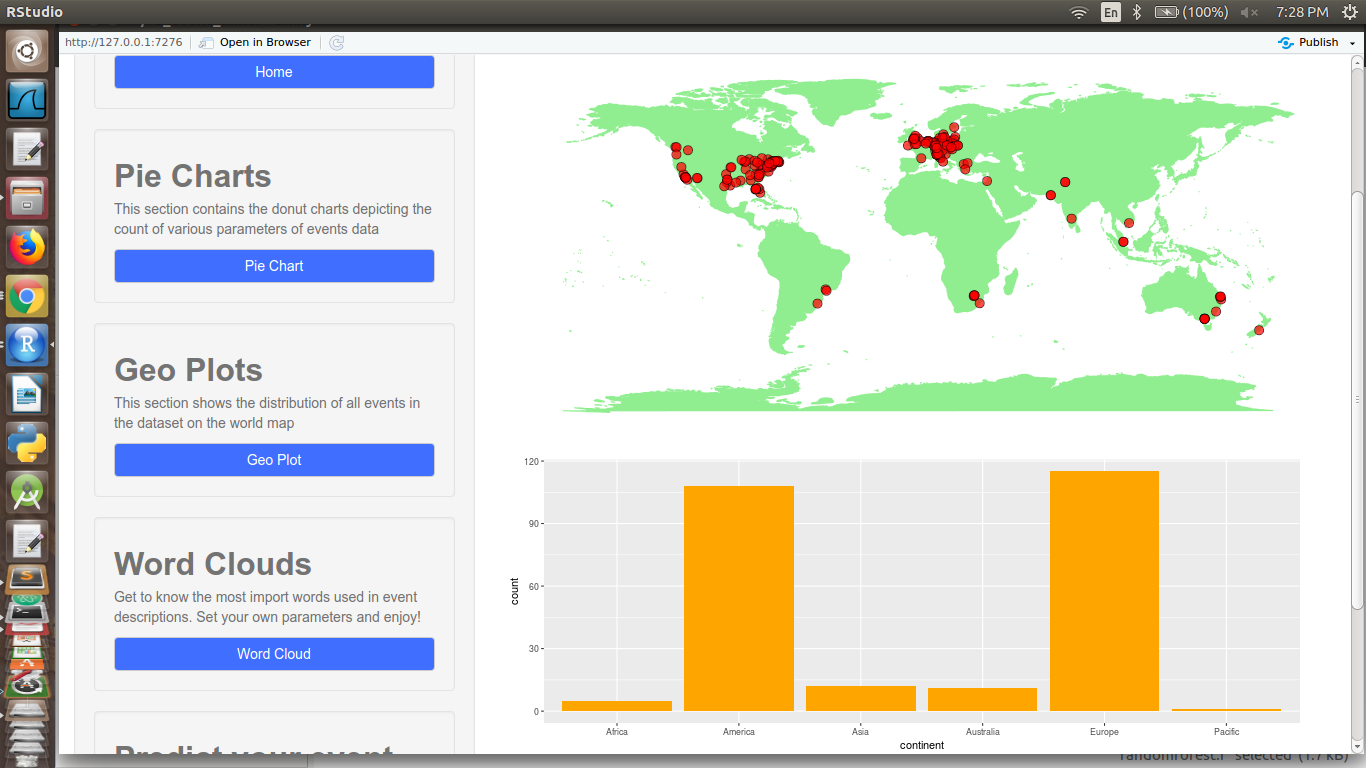
1. **Travel** :



*Fig.9(a): Geo-distribution of travel events .(b) Histogram of continent-wise count of travel events*

Maximum travel events happen in Europe and America.

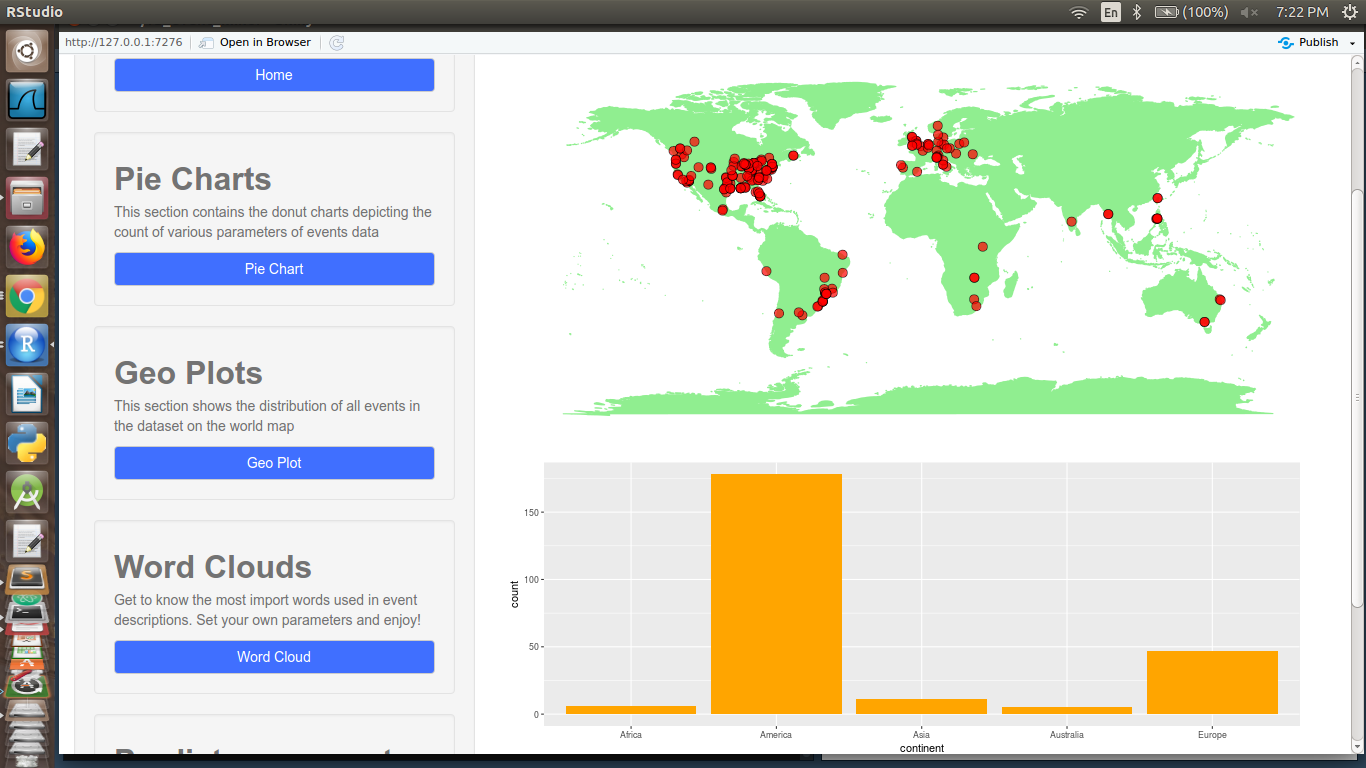
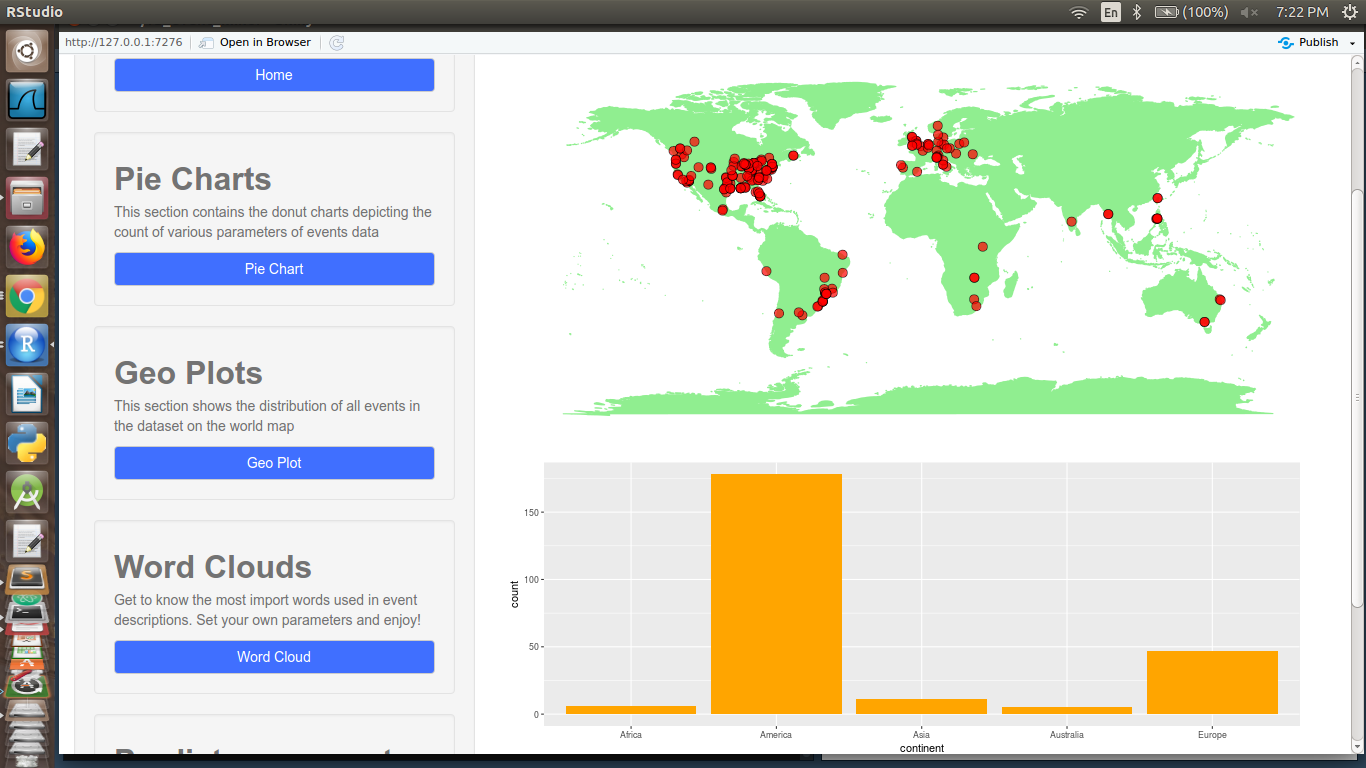
1. **Food:**



*Fig.10(a): Geo-distribution of food events .(b) Histogram of continent-wise count of food events*

Maximum food events happen in Europe and America.

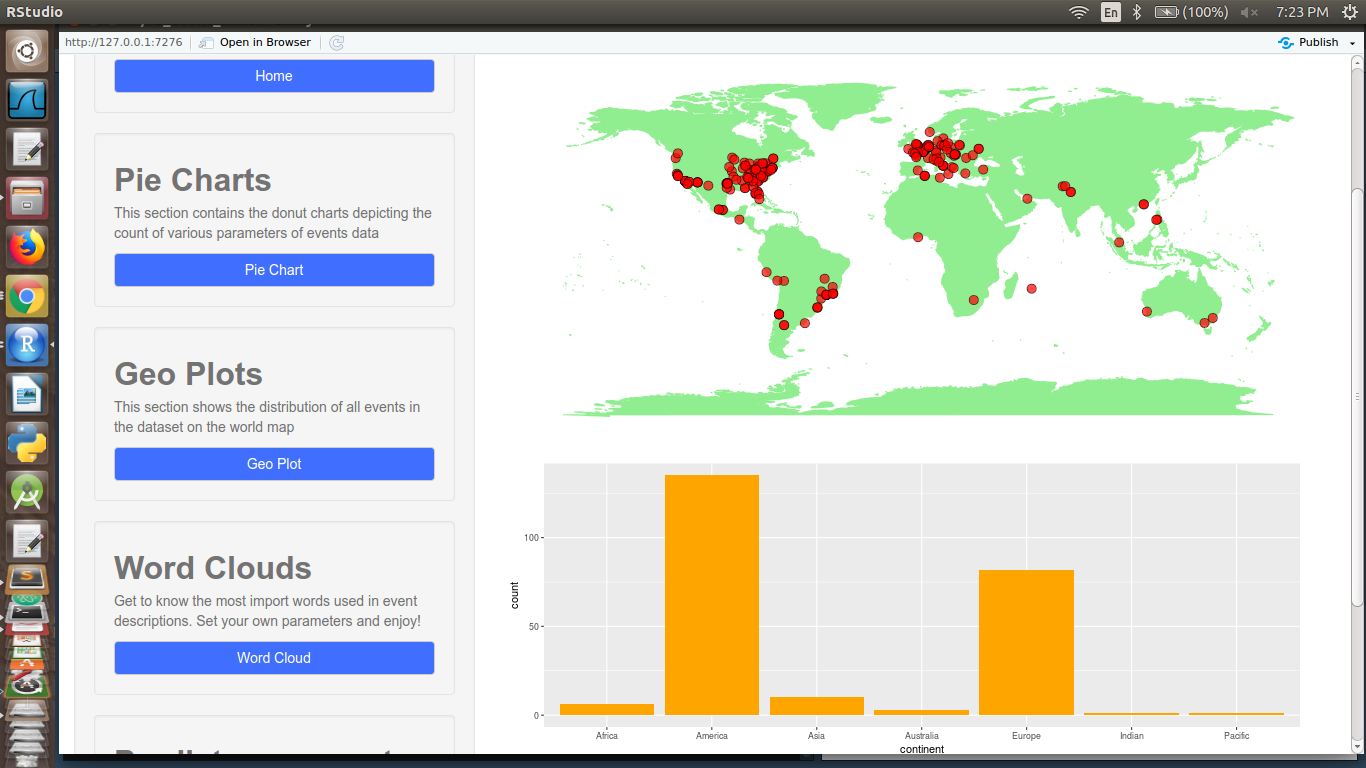
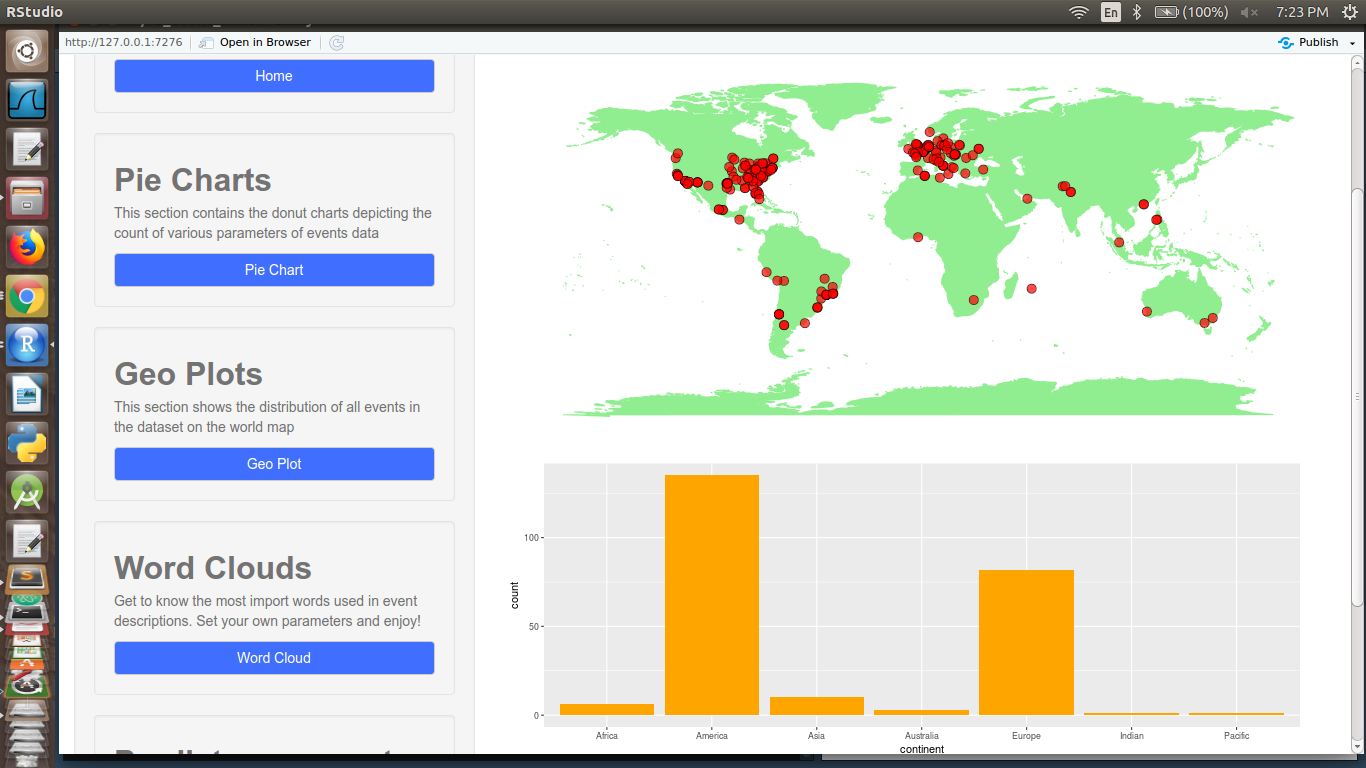
1. **Music:**



*Fig.11(a): Geo-distribution of music events .(b) Histogram of continent-wise count of music events*

Maximum music events happen in America.

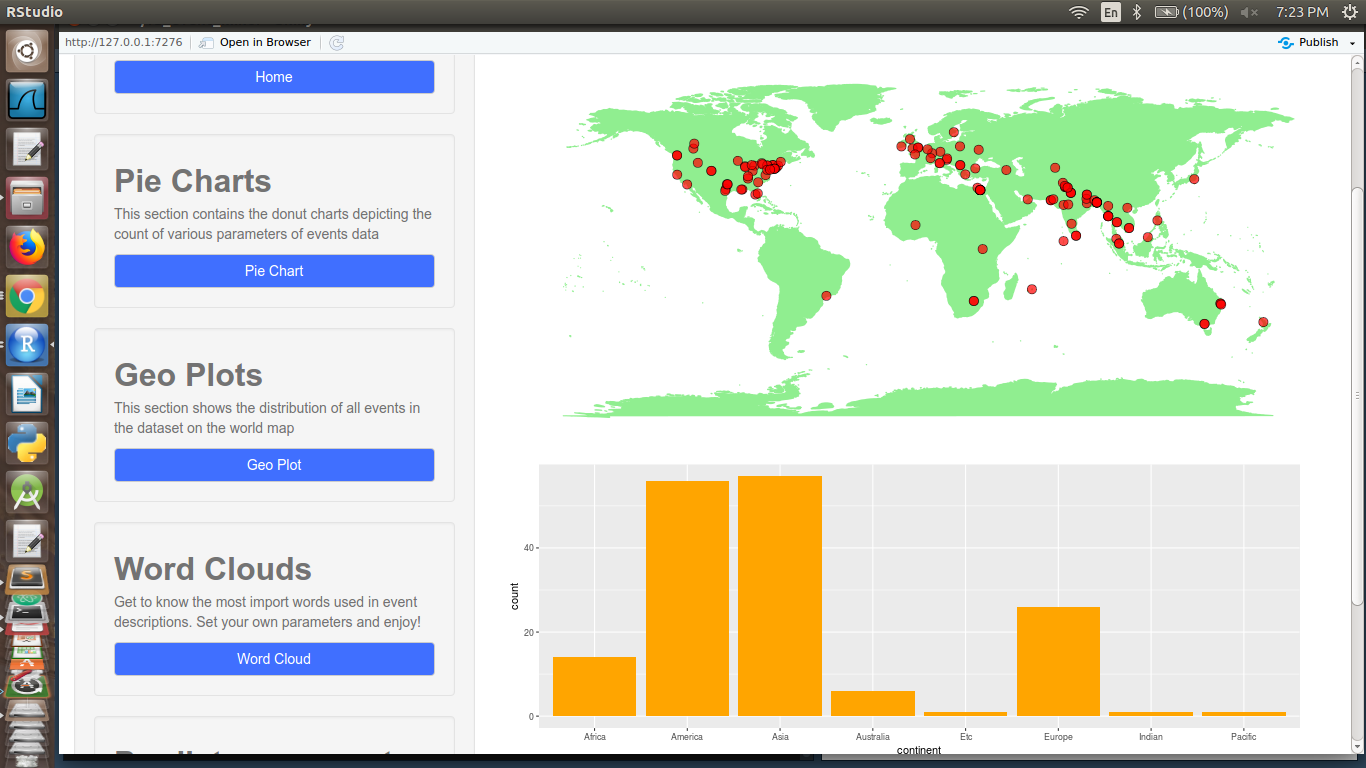
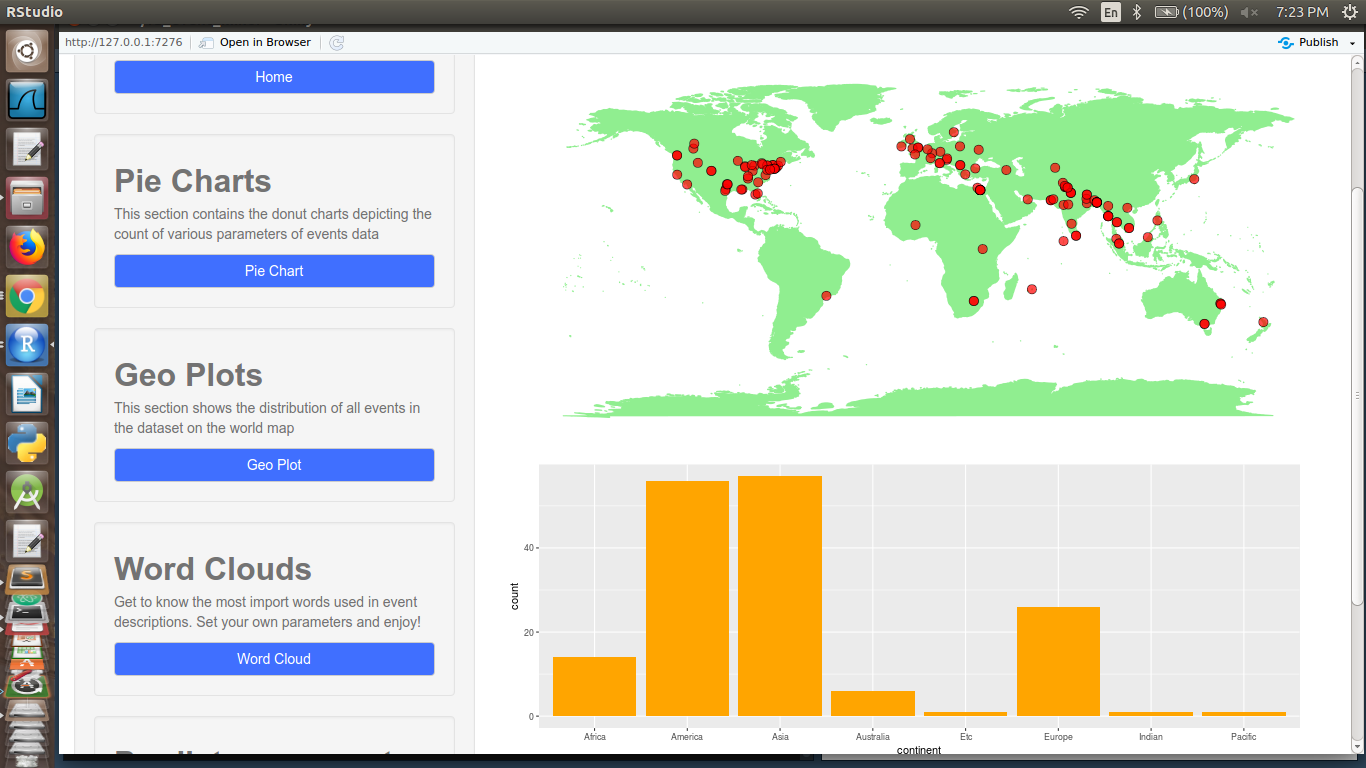
1. **Art:**



*Fig.12(a): Geo-distribution of art events .(b) Histogram of continent-wise count of art events*

Maximum Art events happen in America and Europe.

1. **Education:**



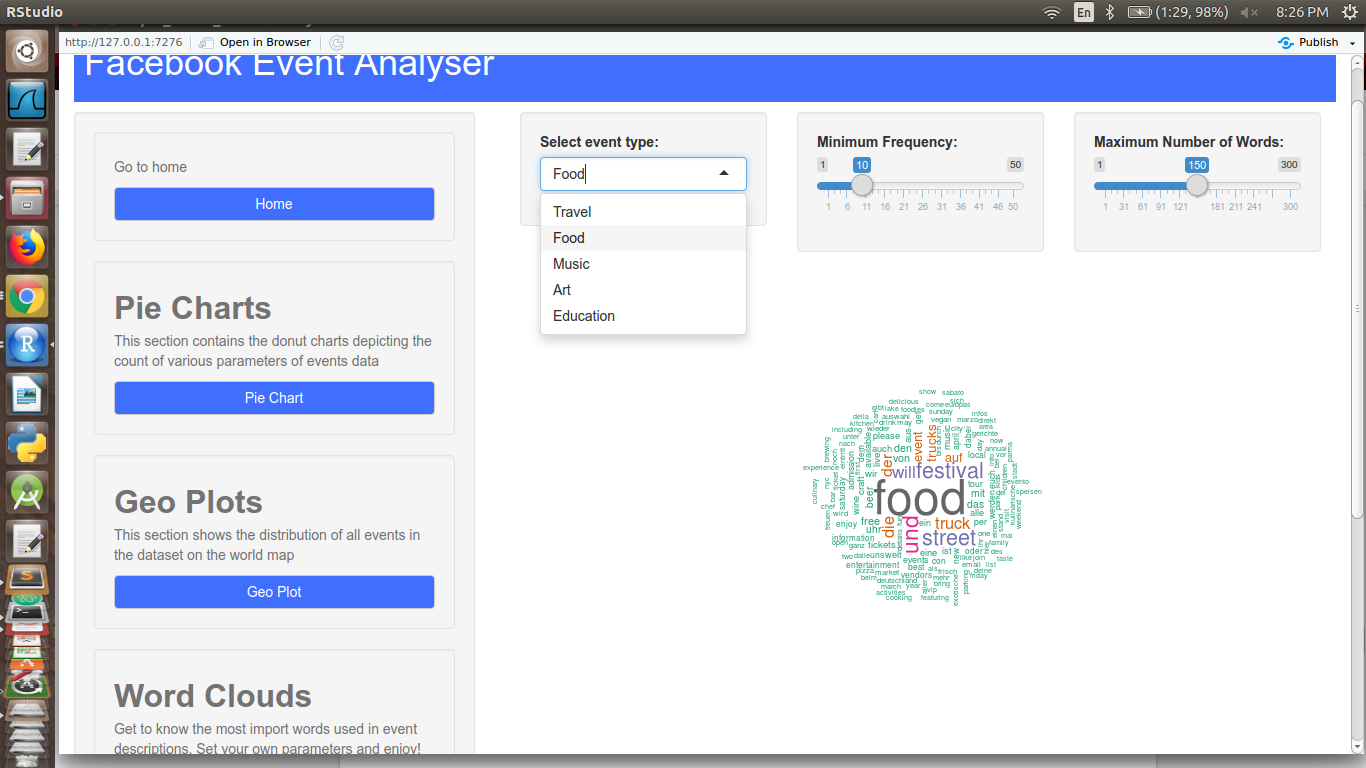
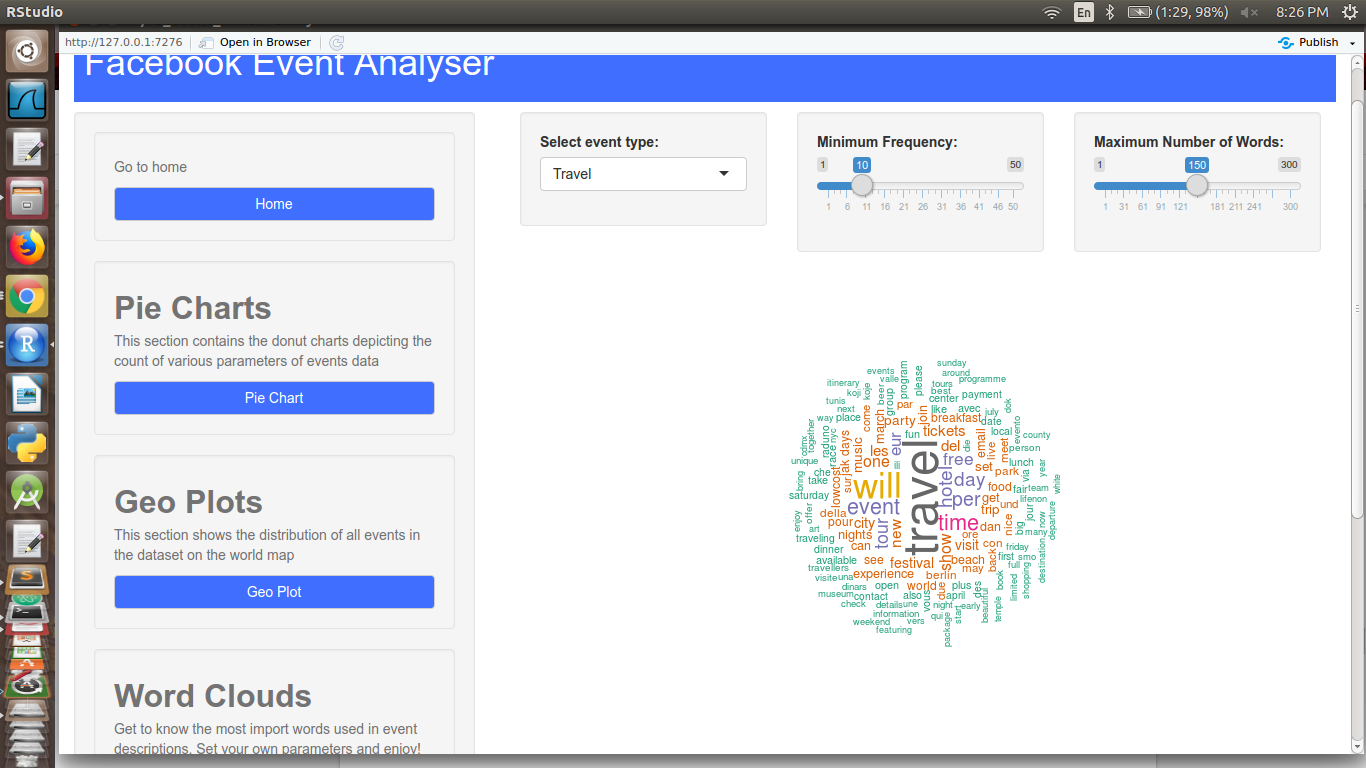
*Fig.13(a): Geo-distribution of education events .(b) Histogram of continent-wise count of education events*

Maximum education events happen in Asia and America.

**WordCloud**

The following word-cloud were produced at minimum word frequency = 10 and maximum number of words = 150

**1.Travel 2. Food**

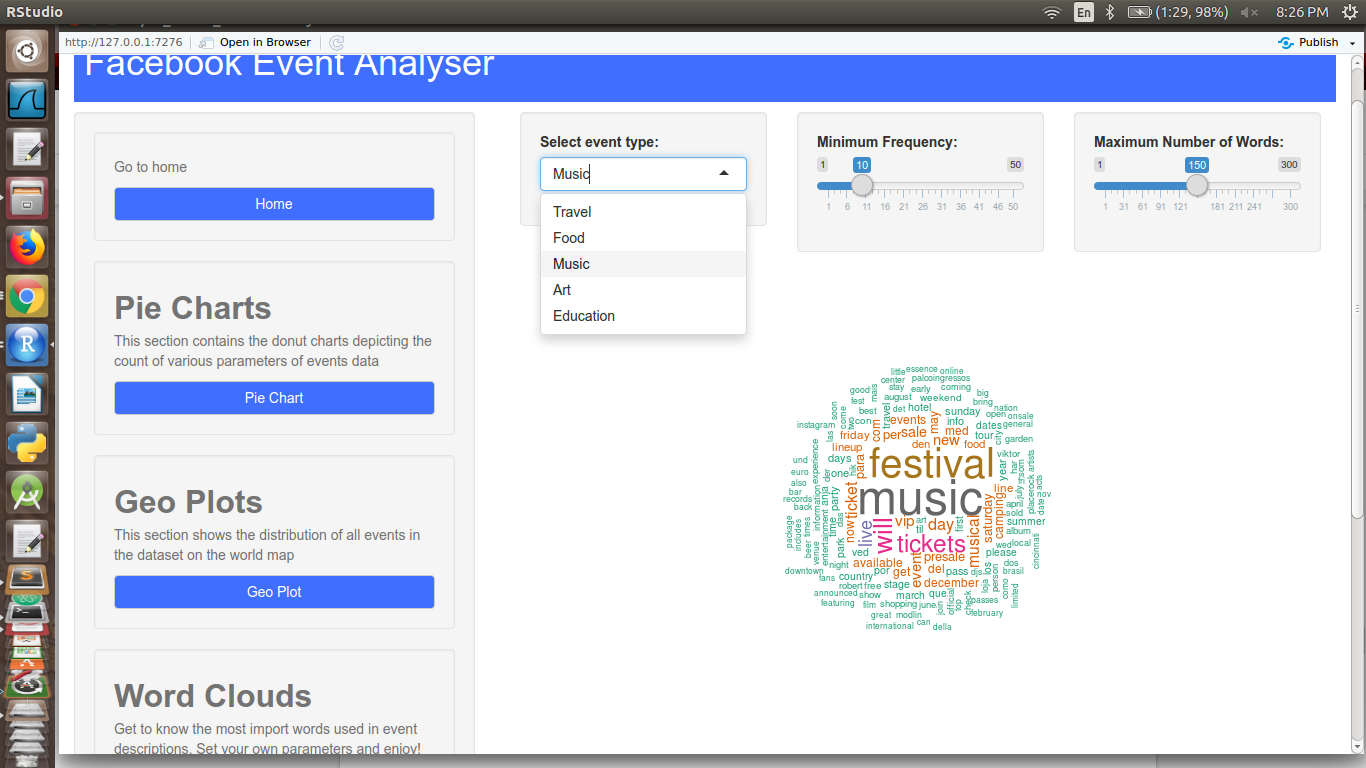
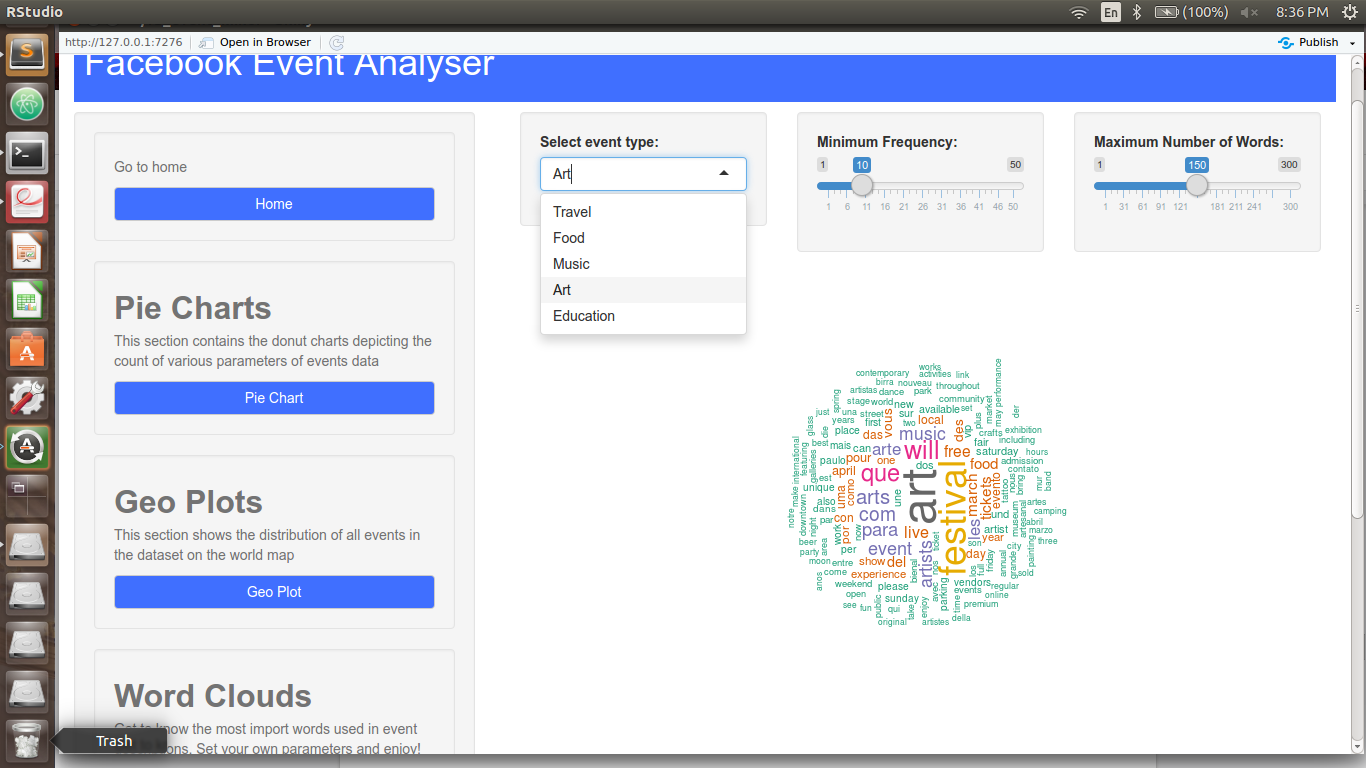


*Fig.14: Word Clouds for (a): travel , (b): Food events*

Top 5 words for travel event description : travel, time, tour, event, hotel

Top 5 words for food event description : food, festival, street, truck, event

**3. Music 4. Art**

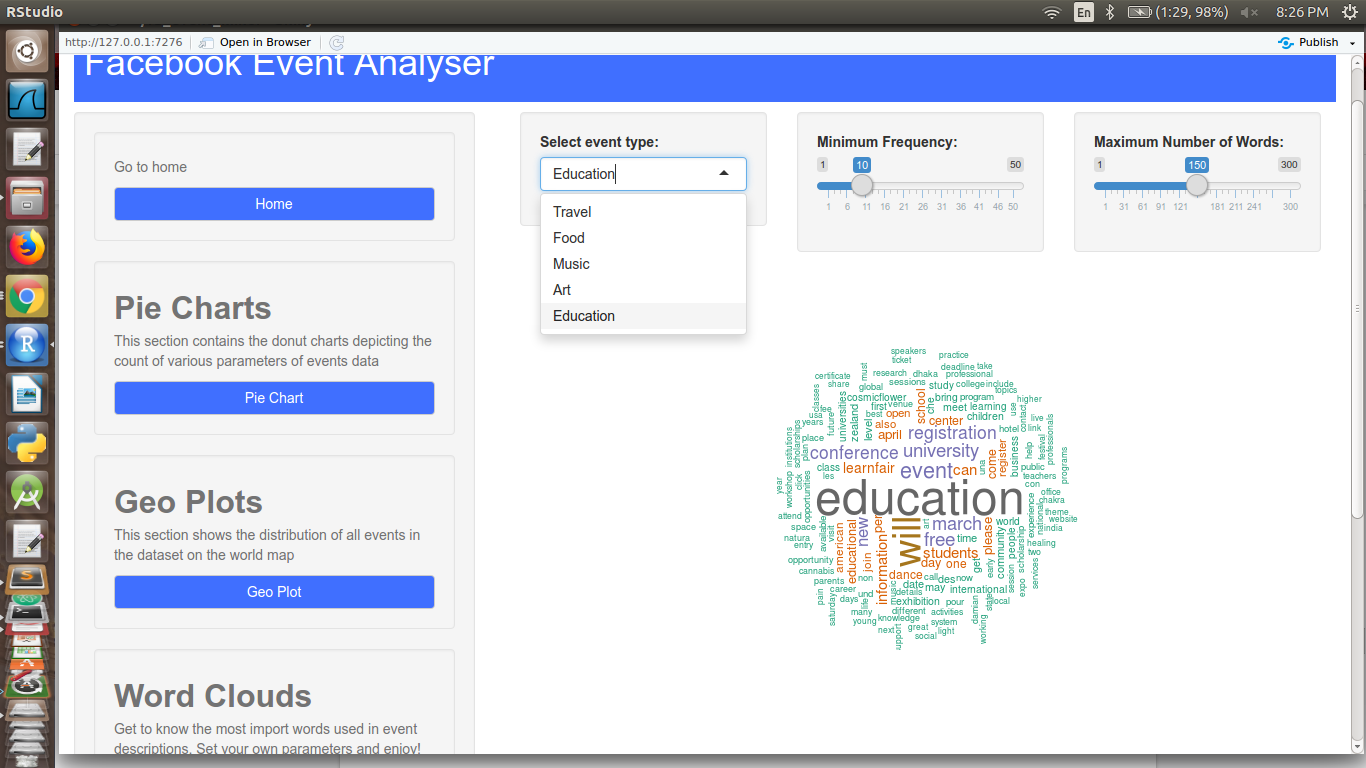
 

*Fig.15: Word Clouds for (a): music , (b): art events*

Top 5 words for music event description : Music, festival, tickets, live, saturday

Top 5 words for art event description : art, festival, event, artist, free

**5. Education**



*Fig.16: Word Clouds for educational events*

Top 5 words for education event description : education, conference, registration, university, student

**Machine Learning**

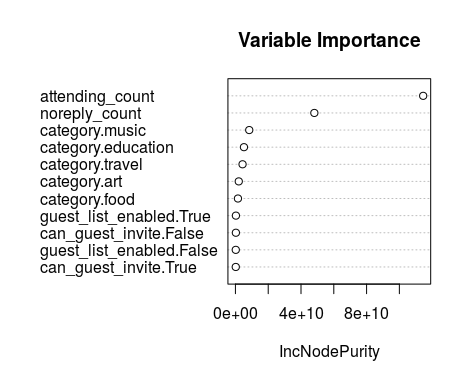
1. **Model : Random Forest**

* Features used :

1. Attending count
2. Maybe count
3. No reply count
4. Can\_guest\_invite
5. Guest\_list\_enabled
6. Category

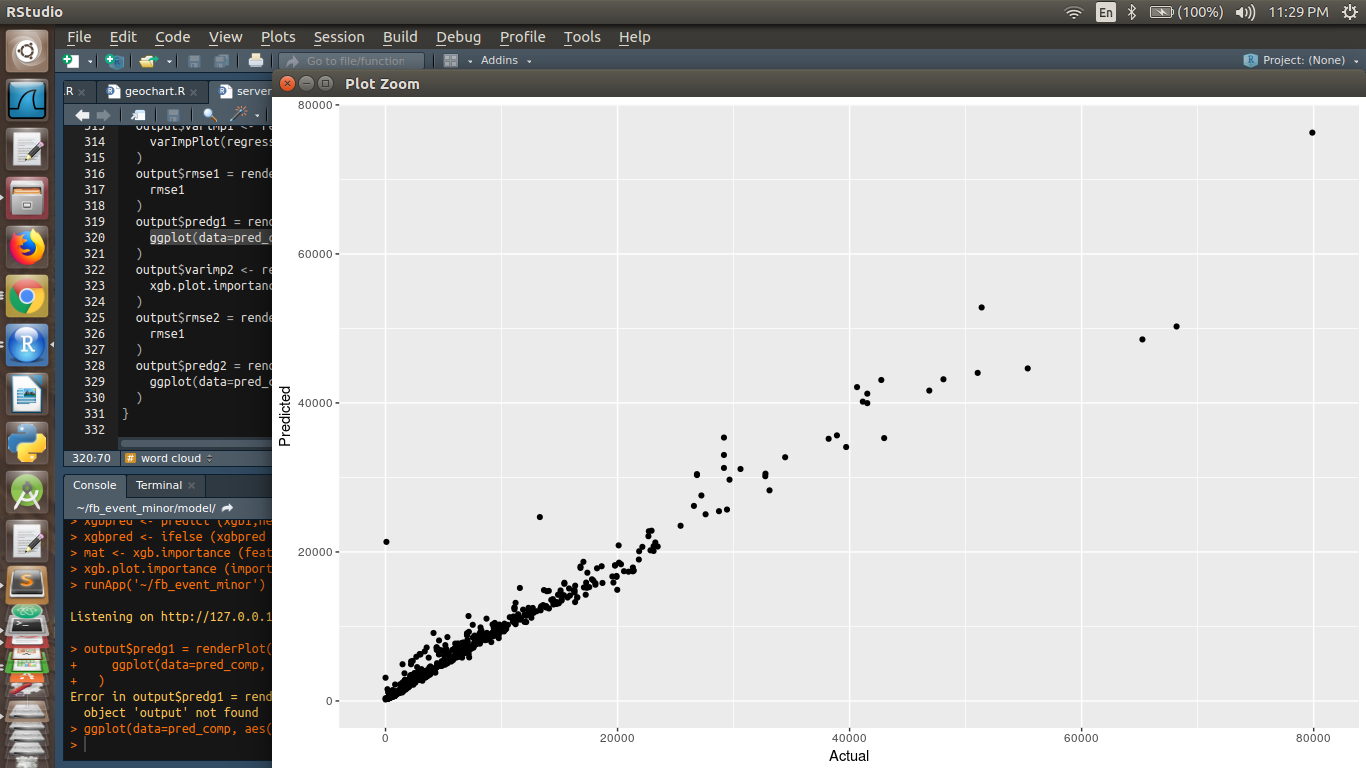
* Split Ratio : 0.75 (75%)
* Parameters : Number of trees = 100
* Results :

1. Accuracy : ~85%
2. Variable Importance :



*Fig. 17: Variable importance graph for random forest model*

1. Actual vs. Predicted graph :



*Fig. 18: The trend between actual and predicted value is linear.*

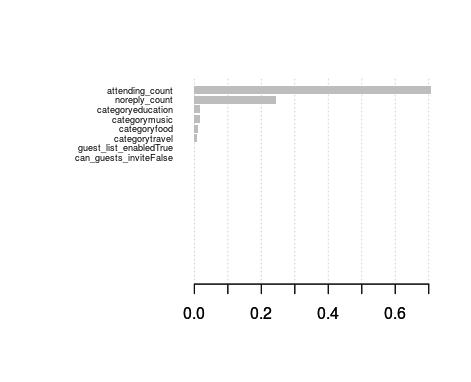
**II. Model : XGBoost**

* Features used :

1. Attending count
2. Maybe count
3. No reply count
4. Can\_guest\_invite
5. Guest\_list\_enabled
6. Category

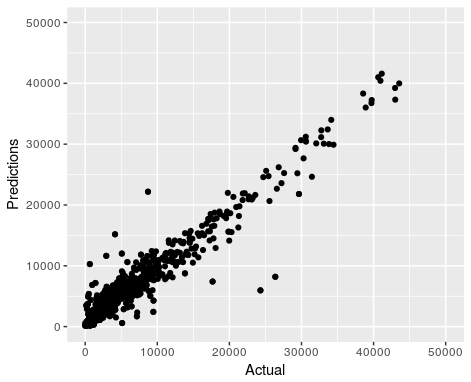
* Split Ratio : 0.75 (75%)
* Parameters: booster=’gbtree’ , objective=’reg:linear’ (linear- regression),eta=0.3(learning rate)
* Result:

1. RMSE- 2549.656
2. Variable Importance :



*Fig. 19 : Variable Importance for XGBoost model*

3. Actual vs. Predicted graph :



*Fig. 20: The trend between actual and predicted value is linear.*

**Conclusion**

We have prepared a tool to analyse effectiveness of various attributes in organizing events like how particular events happening more in particular location and choice of words helping to attract more crowd to event and which type of events attract more crowd.We did our analysis in R programming language in which we used pie- charts, geo-plotting and histogram for plotting our data. For predictive analysis we prepared models like XGboost and random forest to predict number of interested people on basis of some important attributes.We got accuracy of approximate 85% on our test data and also figured out the importance of attributes.The whole analysis was integrated into a web application for which we use a Shiny package in R.

**References**

1. S. Catanese, P. D. Meo, E. Ferrara, G. Fiumara, and A. Provetti, "Crawling Facebook for Social Media Analysis Purposes," in Proc. of the ACM International Conference on Web Intelligence, Mining and Semantics, 2011.

2.Duncan Cleary, "Predictive Analytics in the Public Sector: Using Data Mining to Assist Better Target Selection for Audit", Revenue Irish Tax and Customs Ireland.

3. Ricky Ho, Machine Learning: Patterns for Predictive Analytics.

4.Nann Stefan, Krauss Jonas, Schoder, "Predictive analytics on public data ", Detlef University of Cologne Germany.

5. Shulong Tan, Yang Li, Huan Sun, Ziyu Guan, Xifeng Yan, Jiajun Bu, Chun Chen, Xiaofei He, "Interpreting the Public Sentiment Variations on Facebook", IEEE trans. on knowledge and data engineering, vol. 26, no. 5, May 2014.

6.Using TF-IDF to Determine Word Relevance in Document Queries by Juan Ramos

7.On the selection of decision trees in Random Forests by [Simon Bernard](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin="Authors":.QT.Simon Bernard.QT.&newsearch=true),[Laurent Heutte](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin="Authors":.QT.Laurent Heutte.QT.&newsearch=true),[Sebastien Adam](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin="Authors":.QT.Sebastien Adam.QT.&newsearch=true).

8.‘An analysis of Facebook's graph search’ by [Zubeida Casmod Khan](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin="Authors":.QT.Zubeida Casmod Khan.QT.&newsearch=true),[Thulani Mashiane](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin="Authors":.QT.Thulani Mashiane.QT.&newsearch=true).

9. ‘Word Storms: Multiples of Word Clouds for Visual Comparison of Documents’ by Charles. Suttan.

10. ‘XGBoost: A Scalable Tree Boosting System’ by Tianqi Chen and Carlos Guestrin.

11. Predictive Analytics: A Survey, Trends, Applications, Opportunities & Challenges by Nishchol Mishra, Dr.Sanjay Silakari.