Quantification Of Chatbot Performance Leveraging Exploratory Data Analysis

Presented by

Arnab Dey
21MDT0068

Rupak Dey
21MDT0058

Sudipta Paul
21MDT0042

Under the guidance of
Prof. Rushi Kumar B

OBJECTIVE OF THE PROJECT WORK

With increased usage of chatbots in industries where consumers access personalised information on demand, it requires the businesses to aim at enhancing their chatbot performance for customer satisfaction.

There is a wide range of evaluation frameworks that measures the quality of chatbots using chatbot quality attributes. However, recent studies conclude (Radziwill and Benton, 2017) that an absolute list of quality attributes for evaluating chatbots is non-existent due to high variety of chatbot types. Moreover, assessing a vast majority of quality attributes is time expensive and requires expert's opinion.

The objective of this project is to overcome those challenges by researching what metrics can be automatically measured by analysing chatbot conversations.

ABSTRACT

Chatbots have become an important aspect of industries in our modern times and are gradually coming up as a means of interaction with the consumers. Consumers want access to personalised information when required, preferably 24/7 and in any language. Continuous evaluation of a chatbot performance is crucial for businesses to keep their customers satisfied as it's performance directly influence the user experience (Lemon and Verhoef, 2016)**[1]**.

In general, existing techniques fail to come up with an absolute list of quality attributes for the chatbot performance analysis, but in our report, we proposed a new technique to assess chatbot performance by analysing the chatbot conversations. This enabled us to focus on automatic quantification of chatbot's performance to allow faster prototyping and testing new chatbot models, which requires less costly human evaluations.

INTRODUCTION

In recent years, chatbots have come at the forefront as an important means of communication between consumers and industries where consumers demand for personalised information at any time and in any language. The interaction with a chatbot is ideally supposed to be indistinguishable from that of a human being, but in reality it is not. Therefore, continually ensuring the performance of a chatbot is essential to improve the user-chatbot interaction experience so as to ensure user satisfaction.

There are many existing techniques to assess the performance of chatbots but according to a recent study by Radiziwill and Benton (2017)**[2]** which provides an overview of chatbot quality attributes and assessment frameworks that have been proposed in the last few decades, an absolute list of quality attributes for the assessment purpose does not exist due to the wide variety of chatbot types. It is also difficult and time consuming to properly assess a vast majority of quality attributes.

INTRODUCTION (Contd.)

In our project work, we have proposed a new technique to analyse chatbot conversations and on the basis of the analysis we measure its performance, so as to focus on automating the evaluation process. This will ensure faster, efficient testing and prototyping of new chatbot models.

The purpose of this presentation is to provide an overview of how the entire research project will be performed. Literature review is being presented in Page no. 6, where we discuss the previously proposed techniques and Page no. 8 presents the problem statement, where the research set-up is described by elaborating on the problem statement.

LITERATURE REVIEW

Since chatbots already exist for many years, an extensive number of studies have been previously performed on the topic of chatbot metrics and chatbot evaluation approaches. Contemporary literature is investigated in this study by performing a systematic literature review to identify, critically evaluate and integrate the findings of all relevant, high-quality studies that address one or more research questions (Baumeister & Leary, 1997)**[3]**. The systematic literature review is split up into a systematic literature search followed by a narrative literature review.

To identify relevant literature, a keyword-search is carried out on the scholarly databases: Google Scholar, ResearchGate, WorldCat, and the Computer Science Bibliography (DBLP). The search is performed with the following input:

- Keywords: chatbot, conversational user interface, chatbot AND evaluation, conversational user interface AND evaluation, quantitative evaluation AND chatbot, chatbot quality metrics, conversation metrics, conversation notation standard, chatbot AND sentiment analysis, chatbot metrics, chatbot AND perceived performance;

- Year of publication: 2011 or newer;

LITERATURE REVIEW (Contd.)

Sources published by IEEE, Springer, ACM and Elsevier are preferred, due to their typical state-of-the-art research articles in the field of Computer and Information Sciences.

Radziwill and Benton, 2017, in their paper presented a literature review of quality issues and attributes as they relate to the contemporary issue of chatbot development and implementation. Finally, quality assessment approaches were reviewed, and a quality assessment method based on these attributes and the Analytic Hierarchy Process (AHP) were proposed and examined.

Lemon and Verhoef, 2016, In this article, the authors aim to develop a stronger understanding of customer experience and the customer journey in this era of increasingly complex customer behavior.

Cas Jongerius, 2018, [4] in his paper proposed technique for automatic quantification of chatbot performance by use of metrics.

PROBLEM FORMULATION (STATEMENT OF PROBLEM)

The goal of this project is to create a model, which is capable of analyzing chatbot conversations, to automatically make predictions about the perceived performance of chatbots.

DESIGN APPROACH AND DETAILS

- **Data Collection:** An openly available and large dataset of chat conversations is used in the analysis. We collected the 'Frames' dataset from Maluuba - a Microsoft company - published in 2017 (Asri et al., 2017)**[8]**. The Frames dataset comprises conversations between a user and a chatbot about booking a vacation, including flights and hotels. This dataset was created by letting one human play the role of a chatbot travel agent (wizard) and the other the role of a customer. Dialogues were performed by 12 participants over a period of 20 days. We decided to use the ‘Frames’ dataset, due to its similarities, size and because it contains a survey rating from the users. The dataset is in json format.

DESIGN APPROACH AND DETAILS (Contd.)

❖ **Data Preparation:** The preparation steps are described below:

➢ **Importing Libraries:** Required libraries are imported at first to perform EDA using Python. These included numpy, pandas, matplotlib, seaborn and regular expression**.**

```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
        import re
```

➢ **Reading Data:** We Read the data from the ***json*** file into a pandas dataframe.

```
In [23]: df=pd.read_json("frames.json")
```

DESIGN APPROACH AND DETAILS (Contd.)

The first 5 rows of our original dataset is shown here :

```
In [24]: df.head()
```

Out[24]:

	user_id	turns	wizard_id	id	labels
0	U22HTHYNP	[{'text': 'I'd like to book a trip to Atlantis...	U21DKG18C	e2c0fc6c-2134-4891-8353-ef16d8412c9a	{'userSurveyRating': 4.0, 'wizardSurveyTaskSuc...
1	U21E41CQP	[{'text': 'Hello, I am looking to book a vacat...	U21DMV0KA	4a3bfa39-2c22-42c8-8694-32b4e34415e9	{'userSurveyRating': 3.0, 'wizardSurveyTaskSuc...
2	U21RP4FCY	[{'text': 'Hello there i am looking to go on a...	U21E0179B	6e67ed28-e94c-4fab-96b6-68569a92682f	{'userSurveyRating': 2.0, 'wizardSurveyTaskSuc...
3	U22HTHYNP	[{'text': 'Hi I'd like to go to Caprica from B...	U21DKG18C	5ae76e50-5b48-4166-9f6d-67aaabd7bcaa	{'userSurveyRating': 5.0, 'wizardSurveyTaskSuc...
4	U21E41CQP	[{'text': 'Hello, I am looking to book a trip ...	U21DMV0KA	24603086-bb53-431e-a0d8-1dcc63518ba9	{'userSurveyRating': 5.0, 'wizardSurveyTaskSuc...

➢ **Descriptive Statistics:** We used **describe()** to look at descriptive statistic parameters for the dataset.

```
In [25]: df.describe(include='all')
```

DESIGN APPROACH AND DETAILS (Contd.)

By assigning include attribute a value of ‘all’, we make sure that categorical features are also included in the result. The output Data Frame looks like this:

	user_id	turns	wizard_id	id	labels
count	1369	1369	1369	1369	1369
unique	11	1369	12	1369	16
top	U22K1SX9N	[{'text': 'I'd like to book a trip to Atlantis...	U21T9NMKM	e2c0fc6c-2134-4891-8353-ef16d8412c9a	{'userSurveyRating': 5.0, 'wizardSurveyTaskSuc...
freq	345	1	301	1	929

Since the dataset is unstructured, descriptive statistics was not able to produce numerical values (like mean, standard deviation, percentiles). The categorical features count, unique, top (most frequent value), and corresponding frequency have been populated. From this, we gained an understanding of our dataset.

DESIGN APPROACH AND DETAILS (Contd.)

- **Extraction:** It is necessary to extract the necessary information from the raw unstructured data to a structured dataset before we can carry out EDA. Therefore,
- By using Regular expressions, we were able to find and extract fragments of text that matched the pattern. For further analysis, the extracted fragments were stored in a **csv** file (**Final_Dataset.csv**).

```
x=0
for i in range(1369):
    st=str(df["turns"][i])
    k=st.replace("\'","\"")
    pattern='\"ORIGIN_CITY\":[ a-zA-Z0-9\"\_\.\,\:]+'
    x1=re.findall(pattern,k)
    uid=df["user_id"][i]
    try:
        details=df["turns"][i][0]["labels"]["acts"][1]["args"]
    except Exception as e:
        if not (df["turns"][525][0]["labels"]["acts"][0]):
            date=""
   # print(x)
    #x+=1
    if len(details)!=0:
        for i in details:
            if i["key"]=="str_date":
                date=i["val"]
```

DESIGN APPROACH AND DETAILS (Contd.)

After extraction the data looked as:

In [125]:
```
df=pd.read_csv("Final_Dataset.csv")
df.head()
```

Out[125]:

	Unnamed: 0	ID	Origin	Destination	nadults	Budget	date	price_min	timestamp	num_children
0	0.0	U21RP4FCY	"vancouver"	"naples"	"1"	"3500"	13th of august	"0"	1471283860	"11"
1	1.0	U21RP4FCY	"vancouver"	"naples"	"1"	"4500"	13th of august	"0"	1471284070	"11"
2	2.0	U21RP4FCY	"vancouver"	"naples"	"1"	"4500"	13th of august	"0"	1471284123	"11"
3	3.0	U22HTHYNP	"essen"	"buenos aires"	"1"	"3500"	the 13th	"0"	1471282265	"11"
4	4.0	U21RP4FCY	"mexico city"	"porto alegre"	"1"	"3000.00"	the 13th	"1700"	1471282812	"4"

DESIGN APPROACH AND DETAILS (Contd.)

➢ **Treatments :** After the conversion from unstructured to structured the dataset had to go through some data preprocessing to make it more appropriate and ready to finally gain insights from it. Some of the important treatments we performed were :

- → Removing (“ ”) from the text to make it string or numerical category wise.
- → Extracting Day and Month from the “date” column and storing in new columns.

After the process, the dataset looked like:

```
finaldf.iloc[15:20]
```

	ID	Origin	Destination	month	day	nadults	Budget	price_min	timestamp	num_children
15	U21E41CQP	vancouver	recife	8.0	24	1	NaN	NaN	1471278232	NaN
16	U21E41CQP	vancouver	recife	8.0	24	1	NaN	NaN	1471278238	NaN
17	U22HTHYNP	Cleveland	Milan	NaN	20	3	3000	0	1471358012	3
18	U22HTHYNP	Cleveland	Milan	NaN	20	3	3000	0	1471358072	1
19	U22HTHYNP	Cleveland	Milan	NaN	20	3	5400	0	1471358161	3

DESIGN APPROACH AND DETAILS (Contd.)

- **Missing Value Imputation:** Now, our next step was to check for missing values in our dataset. In those cases where there are any missing entries, we would impute them with appropriate values. Here it shows us how many missing values were present in each column of our dataset.

```
In [37]: finaldf.isnull().sum()

Out[37]: ID              0
         Origin          0
         Destination    66
         month         151
         day            92
         nadults         4
         Budget        319
         price_min     761
         timestamp       0
         num_children  546
         dtype: int64
```

DESIGN APPROACH AND DETAILS (Contd.)

- There were 66 rows having missing values in **'Destination'** column. So, we simply removed those rows. As the total number of rows are 1497, it won't not make any influential effect on our analysis.

```
In [61]: finaldf.dropna(subset=["Destination"],inplace=True)
```

- For the **'month'** column we imputed **8 (Aug)** in half of null values and **9(Sept)** in other half as both of them were the categories in month variable, with more or less equal frequency.

```
In [41]: # 50% Aug. and 50% Sept.
finaldf["month"].fillna("missing",inplace=True)
c=0
for x,i in enumerate(finaldf["month"]):
    if i=="missing" and c<=75:
        finaldf.loc[x,["month"]]=8.0
        c+=1
    elif i=="missing" and c>75:
        finaldf.loc[x,["month"]]=9.0
        c+=1
```

```
In [42]: finaldf["month"].isnull().sum()
Out[42]: 0
```

- For the "**price_min**" column we have replaced the missing values with standard deviation.

DESIGN APPROACH AND DETAILS (Contd.)

- For the **'Day'** column we put **15** in place of null values because we considered the middle of the month.
- In **'nadults',** which corresponds to number of adults in a tour we imputed **3** as the standard/average size of a family is 3. **[5]**
- For **'num_children'** column we imputed **2** in null places as average no. of children of a family is 2. **[6]**

```
In [43]: # day, nadults, num_children imputations
         finaldf["day"].fillna(15,inplace=True)
         finaldf["nadults"].fillna(3,inplace=True)
         finaldf["num_children"].fillna(2,inplace=True)
```

- We imputed average of budget in place of null values of **'Budget'** column as it follows **normal distribution**.

```
In [48]: k.mean()

Out[48]: 0    5724.303905
         dtype: float64

In [62]: finaldf["Budget"].fillna("5724",inplace=True)
```

DESIGN APPROACH AND DETAILS (Contd.)

- **Adding labels:** The goal was to find out how many unique travel routes were searched for that would aid us in further analysis. We have checked the Origin and Destination together for each and every row, if it is different, a unique label will be given, if not then we will check the corresponding user id, if it is different then another unique label will be given else the same label will be repeated. The label starts at **1** and increases gradually up to the last unique route. Now, our dataset is ready for EDA.
- The Final dataset looks like this:

```
df=pd.read_csv("EDA_file.csv")
df.head()
```

	Unnamed: 0	ID	Origin	Destination	month	day	nadults	Budget	price_min	timestamp	num_children	label
0	0	U2709166N	Alexandria	Hiroshima	8	15	1	6700	1	1472657834	1	1
1	1	U2709166N	Alexandria	Hiroshima	8	15	1	1800	2000	1472658012	1	1
2	2	U2709166N	Alexandria	Hiroshima	8	15	1	3200	13000	1472658046	1	1
3	3	U2709166N	Alexandria	Phoenix	8	15	1	5800	13000	1472658095	1	2
4	4	U2709166N	Alexandria	Phoenix	8	15	1	1800	8000	1472658099	1	2

Results

Now since we have our dataset prepared , we will proceed towards the data analysis and visualisation aspect of our work.

Preliminary exploration of the data led us to the following questions that are most appropriate with the context of our work:

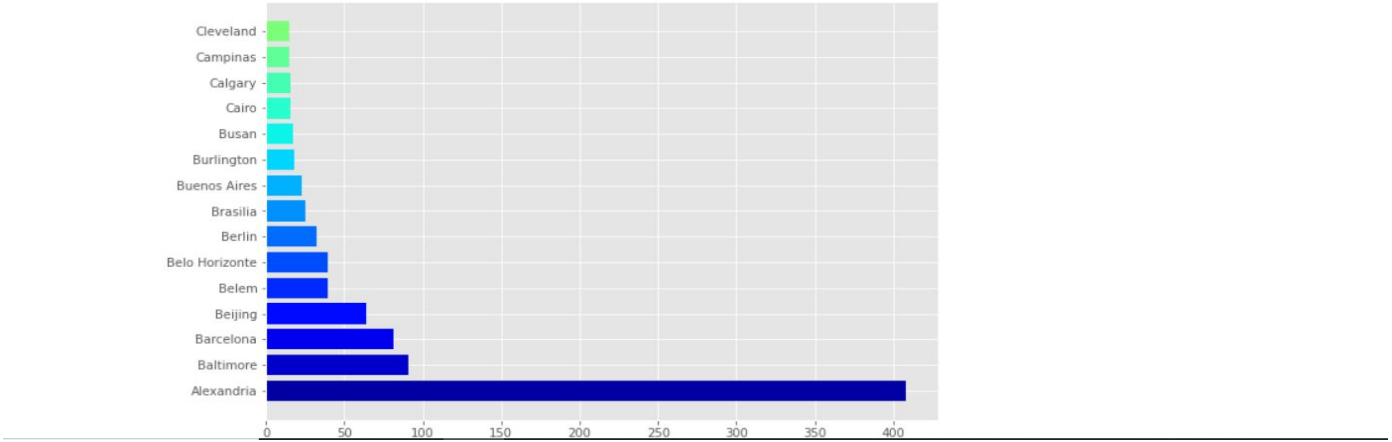
- What is the Most visited city ? (Destination)
- What are the Most important cities from where majority of the travel takes place? (Origin)
- What are the most busiest cities?
- What are some of the important Family friendly places?
- Which cities favour a lot of School or college excursions ?
- What is the most favourable time of the year for travel?
- What is the Most favourable budget?
- Which of the cities have the most luxury tours and average budget tours?
- What is the Minimum price for a particular tour.

Results (Contd.)

To answer the first two questions for identifying the most visited cities and the cities from where majority of travels take place, we use the following **bar plots on Origin** and **Destination** variables to visualise the data and perform univariate analysis as:

In [5]:
```
# Most important city ( Origin )
import matplotlib.cm as cm
from matplotlib.colors import Normalize
plt.style.use('ggplot')
plt.figure(figsize=(15,15))
data = list(range(1,50))
my_cmap = cm.get_cmap('jet')
my_norm = Normalize(vmin=0, vmax=30)
plot1 = plt.subplot2grid((2, 3), (0, 0), colspan=2)
plot1.barh(df['Origin'].unique()[:15], df['Origin'].value_counts()[:15],color=my_cmap(my_norm(data)))
```

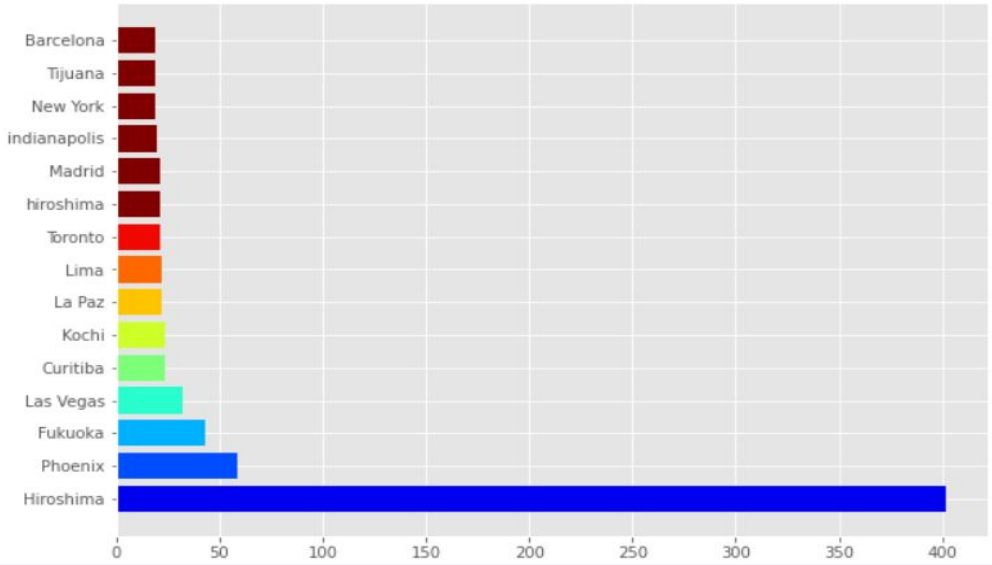
Out[5]: <BarContainer object of 15 artists>



Results (Contd.)

In [6]:
```
# Most visited city ( Destination )
import matplotlib.cm as cm
from matplotlib.colors import Normalize
plt.style.use('ggplot')
plt.figure(figsize=(15,15))
data = list(range(1,35))
my_cmap = cm.get_cmap('jet')
my_norm = Normalize(vmin=0, vmax=10)
plot1 = plt.subplot2grid((2, 3), (0, 0), colspan=2)
plt.barh(df['Destination'].unique()[:15], df['Destination'].value_counts()[:15],color=my_cmap(my_norm(data)))
```

Out[6]: <BarContainer object of 15 artists>



Results (Contd.)

It is observed that **Hiroshima** receives the most number of tourists, while most of the travel takes place from **Alexandria**.

This is further validated by the economically strategic location of the port city of Alexandria in Egypt that facilitates fast travel to Eastern as well as Western hemisphere.

Results (Contd.)

Now coming to the second question of finding which city is busiest in terms of frequency of travels to and from, we observed that Punta Cana comes out at top.

In [12]:
```
l=[]
for i in d1.items():
    if i[1]>=80 and i[1]<=120:
        l.append(i[0])
for j in d2.items():
    if j[1]>=80 and j[1]<=120:
        if j[0] in l:
            print(j[0])
```

```
punta cana
```

Therefore, Punta Cana is the most important tourist place in Caribbean. The Punta Cana International Airport is the primary airport.

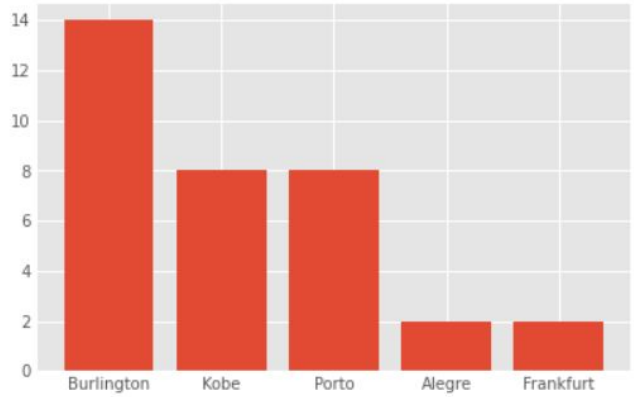
In 2014, Punta Cana received over 2.4 million passengers, making it the second-busiest airport in the Caribbean.

Results (Contd.)

Similarly, for finding out the most family friendly places, we imposed the condition on the data that there has to be **3-4 family members** in total, considering **two parents** and **one or two children.** The result obtained are as follows:

```
In [52]: # Family Friendly places
         dffamily=(df[(df["nadults"]==2) & ((df["num_children"]==1) | (df["num_children"]==2))])
         plt.bar(["Burlington","Kobe","Porto","Alegre","Frankfurt"],dffamily["Destination"].value_counts()[:5])
```

```
Out[52]: <BarContainer object of 5 artists>
```



Results (Contd.)

It is observable that Burlington, Kobe, Porto Alegre and Frankfurt are the cities that tops in searches as the most family friendly places.

This is further justified by real world data and proofs from various surveys.

- According to the **2010 population census,** the population of Porto Alegre is made up of **Roman Catholics (63.85%); Protestants or evangelicals (11.65%); spiritists (7.03%); Umbanda** and **Candomblé (3.35%); the non-religious (10.38%)** and **people of other religions (3.64%).**
- The Church of Jesus Christ of Latter-day Saints has a temple in Porto Alegre. (**Source Wiki**)

Such a city inclined more towards theology and religion points towards the fact that most of the travels it receives are for the family friendly environment that it offers.

Results (Contd.)

Similarly, as quoted in Bing travel website **[7]**,

“ With its intimate atmosphere and age-friendly attractions, Burlington is truly one of the best places to visit in Vermont with kids. Burlington is a small cosmopolitan urban destination with scores of fun activities for the whole family. From the scenic Waterfront Park to pristine beaches,the town oozes with family fun.”

Results (Contd.)

Next, when we focussed on the variable **“num_children”,** which holds the data for number of children per travel, we came across some observations as **40**.

This indicated a huge influx of children in those cities around that time of the year, which indicated towards school or college excursions.

Exploring the data the names of these cities came up:

Results (Contd.)

In [17]:
```
# School or college excursions



df["num_children"].max()
k=df[(df["num_children"]==40)]
k["Destination"].unique()
```

Out[17]:
```
array(['Fukuoka', 'Las Vegas', 'New York', 'Pittsburgh', 'Hamburg',
       'manaus', 'madrid', 'santa cruz', 'osaka'], dtype=object)
```

In [18]:
```
k["Destination"].value_counts()[:5]
```

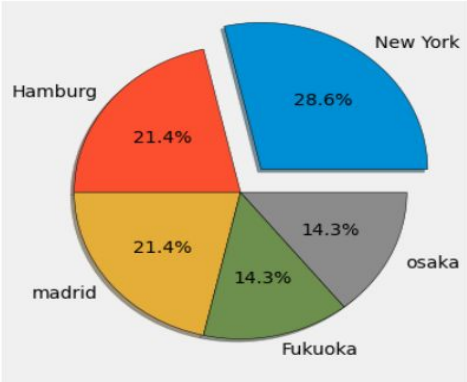
Out[18]:
```
New York    4
Hamburg     3
madrid      3
Fukuoka     2
osaka       2
Name: Destination, dtype: int64
```

Results (Contd.)

On visualization using pie-charts, we got the observations as:

In [19]:
```
plt.style.use('fivethirtyeight')
plt.figure(figsize=(5,10))
l=k["Destination"].value_counts()[:5].keys()
plt.pie(k["Destination"].value_counts()[:5].to_list(),labels=l, explode=[0.2,0,0,0,0], shadow=True, autopct='%1.1f%%', wedgeprops
```

Out[19]:
```
([<matplotlib.patches.Wedge at 0x1e4b27205e0>,
  <matplotlib.patches.Wedge at 0x1e4b272f040>,
  <matplotlib.patches.Wedge at 0x1e4b272f9d0>,
  <matplotlib.patches.Wedge at 0x1e4b273d3a0>,
  <matplotlib.patches.Wedge at 0x1e4b273dd30>],
 [Text(0.8105367016333029, 1.0163809597318352, 'New York'),
  Text(-0.8600146903410576, 0.6858387072756794, 'Hamburg'),
  Text(-0.8600145619153347, -0.68583886831644, 'madrid'),
  Text(0.24477316361895687, -1.072420672297941, 'Fukuoka'),
  Text(0.991065834488076, -0.4772719473323923, 'osaka')],
 [Text(0.49879181638972486, 0.6254652059888216, '28.6%'),
  Text(-0.46909892200421316, 0.37409384033218873, '21.4%'),
  Text(-0.46909885195381884, -0.3740939281726036, '21.4%'),
  Text(0.1335126347012492, -0.5849567303443314, '14.3%'),
  Text(0.5405813642662232, -0.2603301530903958, '14.3%')])
```



Results (Contd.)

From these observations, we drew the inferences that in those cities from where such travels took place educational awareness was high, and the cities to where such tours and excursions happened were centres of academic learning and education.

The cities were **New York, Hamburg,madrid, Fukuoka and Osaka.**

Results (Contd.)

Even the real world data tallied with our assumptions regarding the observations :

FUKUOKA

Nearly ten thousand international students attend universities in or near the Fukuoka prefecture each year. Nearly 200 international conferences are held each year in Fukuoka.(Source : https://en.wikipedia.org/wiki/Fukuoka#Tourism)

Of Japan's eight major cities (excluding metropolitan Tokyo and Osaka), Fukuoka has the second-most university students, with six people in every 100 being students. Kyoto has the highest student ratio. Further, people between 15 and 29 years old make up 25.9% of Fukuoka's population.

(https://survivejapan.weebly.com/fukuoka/10-interesting-facts-about-fukuoka#:~:text=Of%20Japan%E2%80%99s%20eight%20major%20cities,in%20every%20100%20being%20students.)

MADRID

The Autonomous University of Madrid (UAM) is a Spanish public university located in Madrid and founded in 1968. Consistently ranked at the top of the rankings for its teaching and research excellence, the UAM is the Best University in Madrid and the second in Spain according to the QS Ranking 2022 and is among the 400 best universities in the world according to the Academic Ranking of World Universities (ARWU) or Shanghai Ranking 2021.

Hamburg

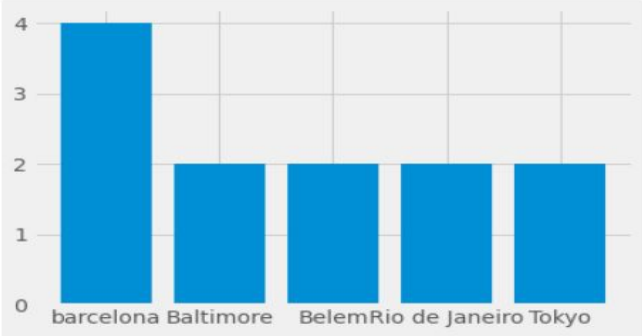
The UNESCO Institute for Lifelong Learning, one of the six educational institutes of UNESCO, is located in Hamburg. (Source Wiki)

Results (Contd.)

Now considering the origin cities of such school excursions, we could obtain the following insights:

```
In [20]: l=k["Origin"].value_counts()[:5].keys()
         plt.bar(l,k["Origin"].value_counts()[:5].to_list())
```

```
Out[20]: <BarContainer object of 5 artists>
```



So we find that there is more educational and career awareness in these 5 cities (Barcelona , Baltimore Belem, Rio de Jjaneiro Tokyo) as there is a huge influx of students from these cities

to the 5 cities of educational excellence found above as New York, Hamburg,madrid, Fukuoka and Osaka.

Even we found that Madrid is closer to Barcelona, so this claim holds true.

Results (Contd.)

Now another interesting aspect that we unearthed from our data is that most of the inter-continental tours were in the month of **August** with frequency of 832 out of the total 1430 searches.

In [21]:
```
# Most favourable time of the year for travel

df["month"].value_counts()

# August and September.
# More preference is August.
```

Out[21]:
```
8    832
9    599
Name: month, dtype: int64
```

Results (Contd.)

And these inter-continental tours were mostly from **Europe and Japan** to **Americas**(Canada and South America)

It may be because **July-August** is the warmest month in Europe. **[8]**

```
In [22]: f=df[(df["month"]==8)][["Origin","Destination"]]
         f["Origin"].value_counts()[:5]

Out[22]: tofino       398
         Mannheim      73
         Tokyo         16
         vancouver     15
         Nagoya        15
         Name: Origin, dtype: int64

In [23]: d=(f["Destination"].value_counts().to_dict())
         d.update({"brasilia":35})
         del d["Brasilia"]

         pd.Series(d)[:5]

Out[23]: sao paulo      401
         Toronto         29
         brasilia        35
         rome            17
         porto alegre    14
         dtype: int64
```

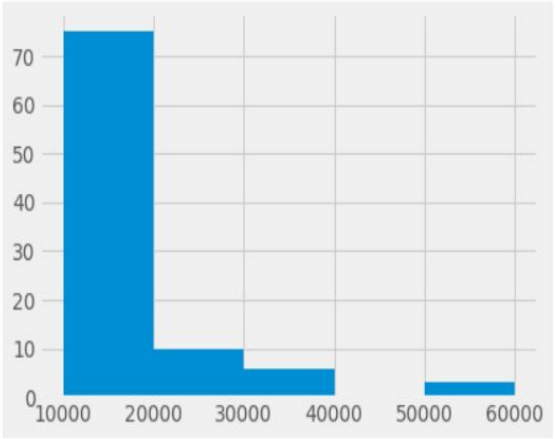
We see that the top 5 origin cities from where most of the travel takes place in the month of August are **European cities**, while the destination cities are in **South America**.

Results (Contd.)

Now from histogram analysis, the most favourable budget seems to be within the range of **"1000-10000" with 5724 (Approx. 6000) being the average.**

```
plt.hist(df["Budget"],bins=[10000,20000,30000,40000,50000,60000])
```

```
Out[26]: (array([75., 10.,  6.,  0.,  3.]),
          array([10000, 20000, 30000, 40000, 50000, 60000]),
          <BarContainer object of 5 artists>)
```



Results (Contd.)

```
In [27]: df["Budget"].value_counts().head(20)

Out[27]: 5724     315
         6000     191
         4500     128
         5000      71
         3000      53
         5800      50
         6100      47
         3800      38
         1200      28
         7400      26
         2300      26
         3700      24
         9700      24
         5600      22
         18600     20
         2700      18
         3600      17
         2500      16
         400       15
         2000      14
         Name: Budget, dtype: int64
```

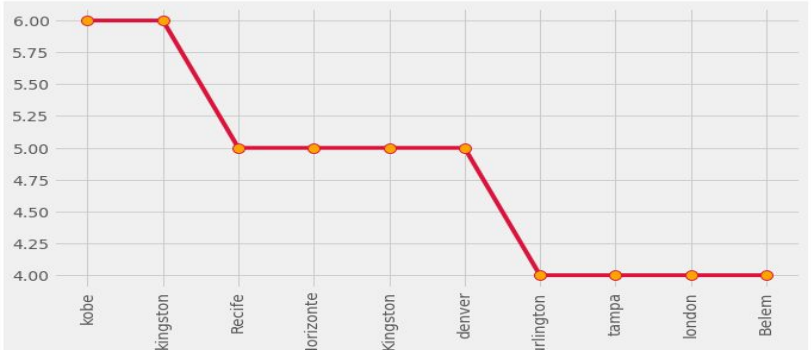
A budget of 5724 appeared for the most number of times.

Results (Contd.)

Now from the frequency of tours with an average budget greater than 10000, we observe that there were cities like **‘tofino’**, **'Mannheim'**, **'punta cana'**, **'Santa Cruz'**, **'Houston'**, **'tijuana'** and **'kobe'**, which suggested posh economy because of the frequency of such luxury tours.

```
In [32]: plt.figure(figsize=(10,5))
         plt.plot(luxury["Origin"].value_counts().keys()[:10],luxury["Origin"].value_counts().to_list()[:10],color='crimson', marker='o',
         plt.xticks(rotation=90)
```

```
Out[32]: ([0, 1, 2, 3, 4, 5, 6, 7, 8, 9],
          [Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, '')])
```



Results (Contd.)

This inferences could be further validated by such real world findings as:

- The Port of Kobe is both an important port and manufacturing center.As of 2004, the city's total real GDP was ¥6.3 trillion, which amounts to thirty-four percent of the GDP for Hyōgo Prefecture and approximately eight percent for the whole Kansai region.
- According to Statistics Canada, the tourism industry in Kingston represents a vital part of the city's economy.

  In 2004, over 3,500 jobs were contributed to Kingston's economy due to the tourism industry

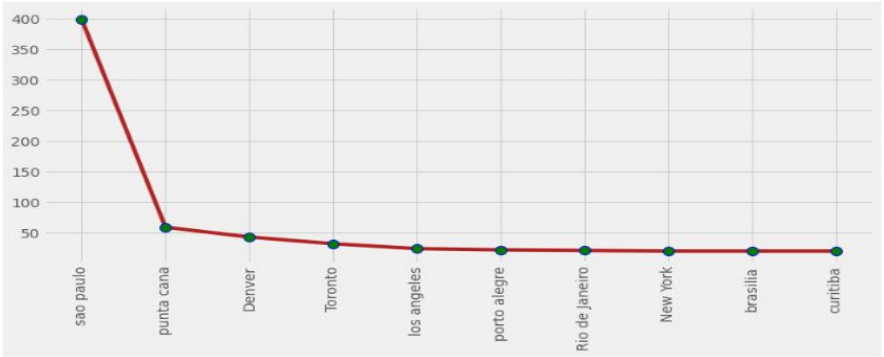
- Belo Horizonte receives large numbers of visitors, as it is in the Brazilian main economic axis, exerting influence even on other states. Multinational and Brazilian companies, such as Google and Oi, maintain offices in the city. The service sector plays a very important role in the economy of Belo Horizonte, being responsible for 85% of the city's Gross Domestic Product (GDP), with other industry making up most of the remaining 15%. Organized FIFA world cup in 2010.

Results (Contd.)

Now, judging by the overall number of tours, with budget around the average budget, these cities suggest that there is an important tourism industry.

```
In [34]: plt.style.use('fivethirtyeight')
         plt.figure(figsize=(12,5))
         plt.plot(tourism["Destination"].value_counts().keys()[:10],tourism["Destination"].value_counts().to_list()[:10], color='firebric
         plt.xticks(rotation=90)
```

```
Out[34]: ([0, 1, 2, 3, 4, 5, 6, 7, 8, 9],
          [Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, ''),
           Text(0, 0, '')])
```



So we conclude that cities like "Sao Paulo , Punta Cana , Denver , Toronto , los angeles" are famous tourist places. Here, tourism is an important industry.

REFERENCES

[1] Lemon, Katherine N., and Peter C. Verhoef. "Understanding customer experience throughout the customer journey." *Journal of marketing* 80.6 (2016): 69-96.

[2]Radziwill, Nicole M., and Morgan C. Benton. "Evaluating quality of chatbots and intelligent conversational agents." *arXiv preprint arXiv:1704.04579* (2017).

[3]Baumeister, Roy F., and Mark R. Leary. "Writing narrative literature reviews." *Review of general psychology* 1.3 (1997): 311-320.

[4] Jongerius, C. M. *Quantifying chatbot performance by using data analytics*. MS thesis. 2018.

[5] https://www.statista.com/statistics/183657/average-size-of-a-family-in-the-us/

[6] Max Roser (2014) - "Fertility Rate". *Published online at OurWorldInData.org*. Retrieved from: 'https://ourworldindata.org/fertility-rate' [Online Resource]

REFERENCES

[7]
https://www.bing.com/ck/a?!&&p=b7eda65990ac85e75cb05235857852d6c1e2d321397195ad9892e47af0bf14eaJmltdHM9MTY1MzMzNzY3NCZpZ3VpZD02OTMzYjMwMC0xNjM5LTRkNzgtYTQ0ZC1kYmZhYTJhNTU1Y2YmaW5zaWQ9NTQwNg&ptn=3&fclid=d346ecf9-dad6-11ec-ba76-ca5e0c10bd30&u=a1aHR0cHM6Ly9mYW1pbHlkZXN0aW5hdGlvbnNndWlkZS5jb20vYmVzdC1mYW1pbHktdmFjYXRpb25zLWluLXZlcm1vbnQvIzp-OnRleHQ9V2l0aCUyMGl0cyUyMGludGltYXRUTIwYXRtb3NwaGVyZSUyMGFuZCUyMGFnZS1mcmllbmRseSUyMGF0dHJhY3Rpb25zJTJDJTIwQnVybGluZ3RvbixwcmlzdGluZSUyMGJlYWNoZXMlMkMlMjB0aGUlMjB0b3duJTIwb296ZXMlMjB3aXRoJTIwZmFtaWx5JTIwZnVuLg&ntb=1

[8]
https://weather-and-climate.com/average-monthly-min-max-Temperature,eu-upper-normandy-fr,France

REFERENCES

[9] https://www.microsoft.com/en-us/research/project/frames-dataset/

THANK YOU