ENHANCING TRAFFIC: EXPLORING GOOGLE APP STORE DATA FOR SOFTWARE HOUSE RECOMMENDATIONS

IN THIS ANALYSIS, WE WILL BE CONDUCTING EXPLORATORY DATA ANALYSIS (EDA) ON THE GOOGLE APP STORE DATASET TO IDENTIFY WHICH APPS SHOULD BE RECOMMENDED TO A SOFTWARE HOUSE TO IMPROVE THEIR CHANCES OF OBTAINING TRAFFIC ACROSS DIFFERENT CATEGORIES.

IMPORTING DATA

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:
```
android_df = pd.read_csv('googleplaystore.csv')
```

CLEANING DATA

In [3]:
```
android_df["Category"].value_counts()
```

Out[3]:
```
FAMILY                 1972
GAME                   1144
TOOLS                   843
MEDICAL                 463
BUSINESS                460
PRODUCTIVITY            424
PERSONALIZATION         392
COMMUNICATION           387
SPORTS                  384
LIFESTYLE               382
FINANCE                 366
HEALTH_AND_FITNESS      341
PHOTOGRAPHY             335
SOCIAL                  295
NEWS_AND_MAGAZINES      283
SHOPPING                260
TRAVEL_AND_LOCAL        258
DATING                  234
BOOKS_AND_REFERENCE     231
VIDEO_PLAYERS           175
EDUCATION               156
ENTERTAINMENT           149
MAPS_AND_NAVIGATION     137
FOOD_AND_DRINK          127
HOUSE_AND_HOME           88
LIBRARIES_AND_DEMO       85
AUTO_AND_VEHICLES        85
WEATHER                  82
ART_AND_DESIGN           65
EVENTS                   64
PARENTING                60
COMICS                   60
BEAUTY                   53
1.9                       1
Name: Category, dtype: int64
```

In [4]:
```
android_df[android_df["Category"]=="1.9"]
```

Out[4]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
10472	Life Made WI-Fi Touchscreen Photo Frame	1.9	19.0	3.0M	1,000+	Free	0	Everyone	NaN	February 11, 2018	1.0.19	4.0 and up	NaN

In [5]:
```
clean_list= ['Life Made WI-Fi Touchscreen Photo Frame', 'LIFESTYLE', 19.0, '3.0M',
       '1,000+', 'Free', '0', 'Everyone', "LIFESTYLE", 'February 11, 2018',
       '1.0.19', '4.0 and up', "nan"]
```

In [6]:
```
android_df[android_df['Category']== "1.9"]= clean_list
```

In [7]:
```
android_df["Category"].value_counts()
```

Out[7]:
```
FAMILY                 1972
GAME                   1144
TOOLS                   843
MEDICAL                 463
BUSINESS                460
PRODUCTIVITY            424
PERSONALIZATION         392
COMMUNICATION           387
SPORTS                  384
LIFESTYLE               383
FINANCE                 366
HEALTH_AND_FITNESS      341
PHOTOGRAPHY             335
SOCIAL                  295
NEWS_AND_MAGAZINES      283
SHOPPING                260
TRAVEL_AND_LOCAL        258
DATING                  234
BOOKS_AND_REFERENCE     231
VIDEO_PLAYERS           175
EDUCATION               156
ENTERTAINMENT           149
MAPS_AND_NAVIGATION     137
FOOD_AND_DRINK          127
HOUSE_AND_HOME           88
AUTO_AND_VEHICLES        85
LIBRARIES_AND_DEMO       85
WEATHER                  82
ART_AND_DESIGN           65
EVENTS                   64
PARENTING                60
COMICS                   60
BEAUTY                   53
Name: Category, dtype: int64
```

In [8]:
```
app_count= android_df["App"].value_counts()
app_count
```

Out[8]:
```
ROBLOX                                               9
CBS Sports App - Scores, News, Stats & Watch Live    8
ESPN                                                 7
Duolingo: Learn Languages Free                       7
Candy Crush Saga                                     7
                                                    ..
Meet U - Get Friends for Snapchat, Kik & Instagram   1
U-Report                                             1
U of I Community Credit Union                        1
Waiting For U Launcher Theme                         1
iHoroscope - 2018 Daily Horoscope & Astrology        1
Name: App, Length: 9660, dtype: int64
```

In [9]:
```
app_count[app_count>1]
```

Out[9]:
```
ROBLOX                                               9
CBS Sports App - Scores, News, Stats & Watch Live    8
ESPN                                                 7
Duolingo: Learn Languages Free                       7
Candy Crush Saga                                     7
                                                    ..
Transenger – Ts Dating and Chat for Free             2
Random Video Chat                                    2
Clover Dating App                                    2
Docs To Go™ Free Office Suite                        2
English Dictionary - Offline                         2
Name: App, Length: 798, dtype: int64
```

In [10]:
```
"Instagram" in app_count[app_count>1].index
```

Out[10]: True

In [11]:
```
android_df[android_df["App"]=="Instagram"]
```

Out[11]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
2545	Instagram	SOCIAL	4.5	66577313	Varies with device	1,000,000,000+	Free	0	Teen	Social	July 31, 2018	Varies with device	Varies with device
2604	Instagram	SOCIAL	4.5	66577446	Varies with device	1,000,000,000+	Free	0	Teen	Social	July 31, 2018	Varies with device	Varies with device
2611	Instagram	SOCIAL	4.5	66577313	Varies with device	1,000,000,000+	Free	0	Teen	Social	July 31, 2018	Varies with device	Varies with device
3909	Instagram	SOCIAL	4.5	66509917	Varies with device	1,000,000,000+	Free	0	Teen	Social	July 31, 2018	Varies with device	Varies with device

In [12]:
```
duplicate_apps_df = android_df[android_df.duplicated(subset=["App"], keep= False)]
duplicate_apps_df[duplicate_apps_df["App"] == "Instagram"]
```

Out[12]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
2545	Instagram	SOCIAL	4.5	66577313	Varies with device	1,000,000,000+	Free	0	Teen	Social	July 31, 2018	Varies with device	Varies with device
2604	Instagram	SOCIAL	4.5	66577446	Varies with device	1,000,000,000+	Free	0	Teen	Social	July 31, 2018	Varies with device	Varies with device
2611	Instagram	SOCIAL	4.5	66577313	Varies with device	1,000,000,000+	Free	0	Teen	Social	July 31, 2018	Varies with device	Varies with device
3909	Instagram	SOCIAL	4.5	66509917	Varies with device	1,000,000,000+	Free	0	Teen	Social	July 31, 2018	Varies with device	Varies with device

In [13]:
```
num_duplicate_app = duplicate_apps_df["App"].nunique()
```

In [14]:
```
reviews_max = android_df.groupby("App")['Reviews'].max()
```

In [15]:
```
reviews_max["Instagram"]
```

Out[15]: '66577446'

REMOVING DUPPLICATES REVIEWS

In [16]:
```
android_clean = []
already_added = []
for index, row in android_df.iterrows():
    name = row["App"]
    n_reviews = row["Reviews"]
    if (reviews_max[name]== n_reviews) and (name not in already_added):
        android_clean.append(row)
        already_added.append(name)
```

In [17]:
```
len(android_clean)
```

Out[17]: 9660

In [18]:
```
android_clean = pd.DataFrame(android_clean)
```

In [19]:
```
android_clean
```

Out[19]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Curre V
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Vari wi devi
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1
5	Paper flowers instructions	ART_AND_DESIGN	4.4	167	5.6M	50,000+	Free	0	Everyone	Art & Design	March 26, 2017	1
...	...	...	...	...	...	...	...	...	...	...	...	...
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53M	5,000+	Free	0	Everyone	Education	July 25, 2017	1.4
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3.6M	100+	Free	0	Everyone	Education	July 6, 2018	1
10838	Parkinson Exercices FR	MEDICAL	NaN	3	9.5M	1,000+	Free	0	Everyone	Medical	January 20, 2017	1
10839	The SCP Foundation DB fr nn5n	BOOKS_AND_REFERENCE	4.5	114	Varies with device	1,000+	Free	0	Mature 17+	Books & Reference	January 19, 2015	Vari wi devi
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19M	10,000,000+	Free	0	Everyone	Lifestyle	July 25, 2018	Vari wi devi

9660 rows × 13 columns

REMOVING NON ENGLISH APPS FROM DATA

In [20]:
```
def is_english(app_name):
    lst = []
    for i in app_name:
        if ord(i)>127:
            lst.append(False)
        else:
            lst.append(True)
    non_ascii = 0
    for j in lst:
        if j == False:
            non_ascii += 1
    if non_ascii>3:
        return False
    else:
        return True
```

In [21]:
```
android_english = android_clean[android_clean["App"].apply(is_english)]
```

In [22]:
```
android_english.shape
```

Out[22]: (9615, 13)

In [23]:
```
android_final = android_english[android_english["Price"]=="0"]
android_final.shape
```

Out[23]: (8862, 13)

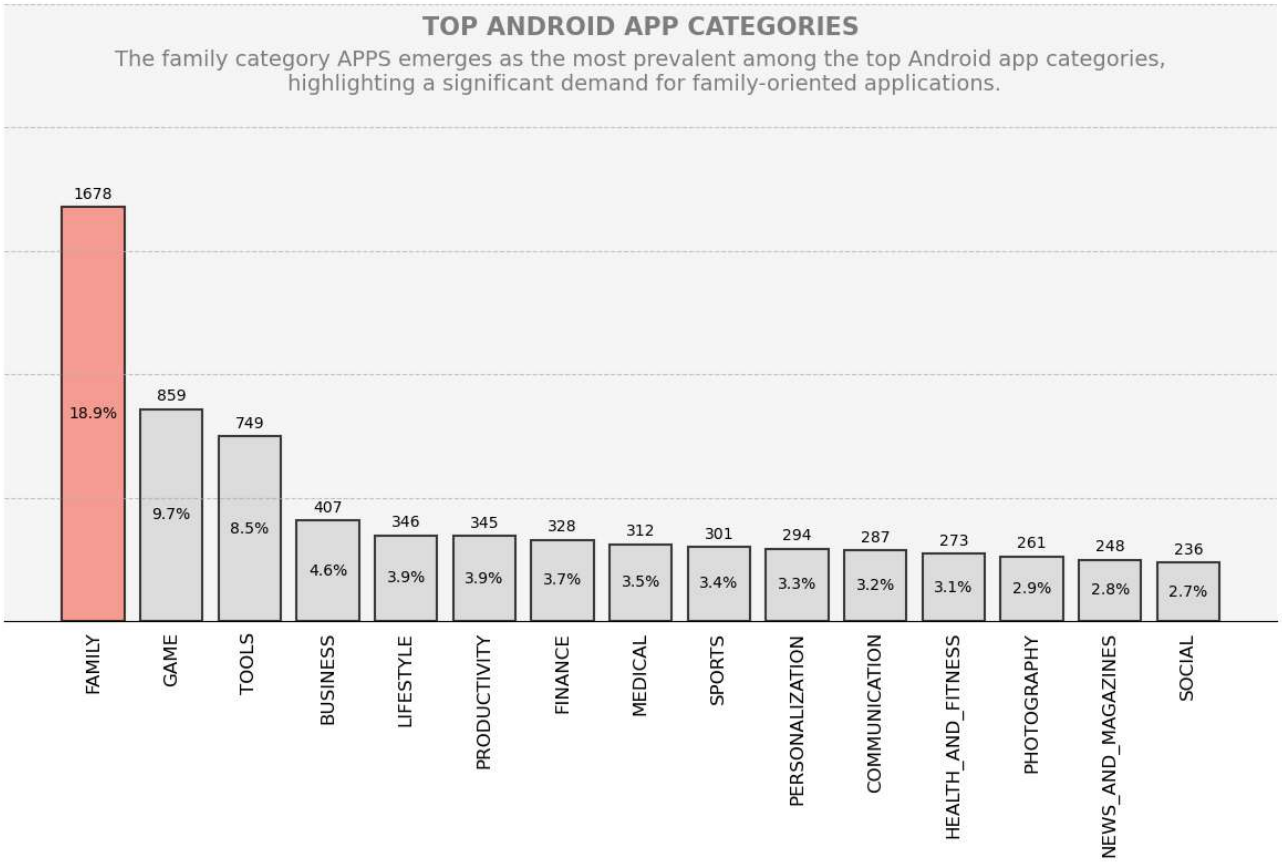
In [24]:
```
android_final["Category"].value_counts(normalize = True)*100
```

Out[24]:
```
FAMILY                 18.934778
GAME                    9.693072
TOOLS                   8.451817
BUSINESS                4.592643
LIFESTYLE               3.904311
PRODUCTIVITY            3.893026
FINANCE                 3.701196
MEDICAL                 3.520650
SPORTS                  3.396524
PERSONALIZATION         3.317536
COMMUNICATION           3.238547
HEALTH_AND_FITNESS      3.080569
PHOTOGRAPHY             2.945159
NEWS_AND_MAGAZINES      2.798465
SOCIAL                  2.663056
TRAVEL_AND_LOCAL        2.335816
SHOPPING                2.245543
BOOKS_AND_REFERENCE     2.143986
DATING                  1.861882
VIDEO_PLAYERS           1.794177
MAPS_AND_NAVIGATION     1.399233
FOOD_AND_DRINK          1.241255
EDUCATION               1.173550
ENTERTAINMENT           0.959151
LIBRARIES_AND_DEMO      0.936583
AUTO_AND_VEHICLES       0.925299
HOUSE_AND_HOME          0.823742
WEATHER                 0.801174
EVENTS                  0.710900
PARENTING               0.654480
ART_AND_DESIGN          0.643196
COMICS                  0.620627
BEAUTY                  0.598059
Name: Category, dtype: float64
```

CREATING GRAPH FOR "TOP APP CATEGORIES"

In [25]:
```
categories = android_final["Category"].value_counts().index[:15]
counts = android_final['Category'].value_counts().values[:15]
percentage = round(android_final['Category'].value_counts(normalize = True)*100,1)[:15]

plt.figure(figsize = (12,8))
bars = plt.bar(categories, counts, color = "lightgray", alpha=0.75, edgecolor= "black", linewidth=1.5)
plt.xticks(rotation=90, fontsize=12)
plt.yticks(fontsize=12)
plt.grid(axis="y",linestyle="--", alpha=0.7)
plt.grid(axis="x",linestyle="")
plt.xticks(fontsize=12)
plt.yticks(range(0,3000,500),[], fontsize=12)
plt.tick_params(bottom = 0, left = 0)
max_count_category= categories[counts.argmax()]
max_count_index= list(categories).index(max_count_category)
bars[max_count_index].set_color("salmon")
bars[max_count_index].set_edgecolor("black")
for bar , prec in zip(bars, percentage):
    height = bar.get_height()
    plt.text(bar.get_x()+bar.get_width()/2, height +20, '%d' % int(height), ha= 'center', va='bottom', fontsize=
    plt.text(bar.get_x()+bar.get_width()/2, height/2, f'{prec}%', ha= 'center', va='center', fontsize=10, color=
ax = plt.gca()
ax.set_facecolor('#f7f7f7')
plt.text(0.5,0.95,'TOP ANDROID APP CATEGORIES', horizontalalignment= 'center', fontsize=16, transform=plt.gca().
plt.text(0.5,0.86,'The family category APPS emerges as the most prevalent among the top Android app categories,\r
for i in ["top","right","left"]:
    plt.gca().spines[i].set_visible(False)
plt.tight_layout()
plt.savefig('TOP CATAGORY APPS.jpg', format='jpeg')
plt.show()
```



PREPARING DATA FOR ANALYSING APPS ON THE BASES OF INSTALLATIONS

In [26]:
```
android_final[android_final["Category"]=="FAMILY"]
```

Out[26]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
2017	Jewels Crush- Match 3 Puzzle	FAMILY	4.4	14774	19M	1,000,000+	Free	0	Everyone	Casual;Brain Games	July 23, 2018	1.9.3901	4.0.3 and up
2018	Coloring & Learn	FAMILY	4.4	12753	51M	5,000,000+	Free	0	Everyone	Educational;Creativity	July 17, 2018	1.49	4.0.3 and up
2019	Mahjong	FAMILY	4.5	33983	22M	5,000,000+	Free	0	Everyone	Puzzle;Brain Games	August 2, 2018	1.24.3181	4.0.3 and up
2020	Super ABC! Learning games for kids! Preschool ...	FAMILY	4.6	20267	46M	1,000,000+	Free	0	Everyone	Educational;Education	July 16, 2018	1.1.6.7	4.1 and up
2021	Toy Pop Cubes	FAMILY	4.5	5761	21M	1,000,000+	Free	0	Everyone	Casual;Brain Games	July 4, 2018	1.8.3181	4.0.3 and up
...	...	...	...	...	...	...	...	...	...	...	...	...	...
10821	Poop FR	FAMILY	NaN	6	2.5M	50+	Free	0	Everyone	Entertainment	May 29, 2018	1.0	4.0.3 and up
10827	Fr Agnel Ambarnath	FAMILY	4.2	117	13M	5,000+	Free	0	Everyone	Education	June 13, 2018	2.0.20	4.0.3 and up
10834	FR Calculator	FAMILY	4.0	7	2.6M	500+	Free	0	Everyone	Education	June 18, 2017	1.0.0	4.1 and up
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53M	5,000+	Free	0	Everyone	Education	July 25, 2017	1.48	4.1 and up
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3.6M	100+	Free	0	Everyone	Education	July 6, 2018	1.0	4.1 and up

1678 rows × 13 columns

In [27]:
```
android_final["Installs"].value_counts(normalize = True )*100
```

Out[27]:
```
1,000,000+        15.741368
100,000+          11.554954
10,000,000+       10.516813
10,000+           10.200858
1,000+             8.395396
100+               6.917174
5,000,000+         6.838186
500,000+           5.574362
50,000+            4.773189
5,000+             4.513654
10+                3.543218
500+               3.249831
50,000,000+        2.290679
100,000,000+       2.121417
50+                1.918303
5+                 0.789889
1+                 0.507786
500,000,000+       0.270819
1,000,000,000+     0.225683
0+                 0.045137
0                  0.011284
Name: Installs, dtype: float64
```

CLEANING AND ADDING SUFFIXES FOR BETTER VISUALIZATION

In [28]:
```
android_final["Installs_int"]= android_final["Installs"].str.replace(",","").str.replace("+","").astype(int)
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_6264\2898442273.py:1: FutureWarning: The default value of regex will c
hange from True to False in a future version. In addition, single character regular expressions will *not* be t
reated as literal strings when regex=True.
  android_final["Installs_int"]= android_final["Installs"].str.replace(",","").str.replace("+","").astype(int)
C:\Users\hp\AppData\Local\Temp\ipykernel_6264\2898442273.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret
urning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-
view-versus-a-copy)
  android_final["Installs_int"]= android_final["Installs"].str.replace(",","").str.replace("+","").astype(int)
```

In [29]:
```
install_frq = android_final["Installs_int"].value_counts().sort_index()
install_frq = install_frq[install_frq.index>500]
install_frq
```

Out[29]:
```
1000           744
5000           400
10000          904
50000          423
100000        1024
500000         494
1000000       1395
5000000        606
10000000       932
50000000       203
100000000      188
500000000       24
1000000000      20
Name: Installs_int, dtype: int64
```

In [30]:
```
install_frq_per = round(android_final["Installs_int"].value_counts(normalize=True)*100,2).sort_index()
install_frq_per = install_frq_per[install_frq_per.index>500]
install_frq_per
```

Out[30]:
```
1000           8.40
5000           4.51
10000         10.20
50000          4.77
100000        11.55
500000         5.57
1000000       15.74
5000000        6.84
10000000      10.52
50000000       2.29
100000000      2.12
500000000      0.27
1000000000     0.23
Name: Installs_int, dtype: float64
```

In [31]:
```
def alphanum_units(value):
    if value >=1e9:
        return f'{value/1e9:.0f}B'
    elif value >=1e6:
        return f'{value/1e6:.0f}M'
    elif value >=1e3:
        return f'{value/1e3:.0f}K'
    else:
        return f'{value:.0f}'
```

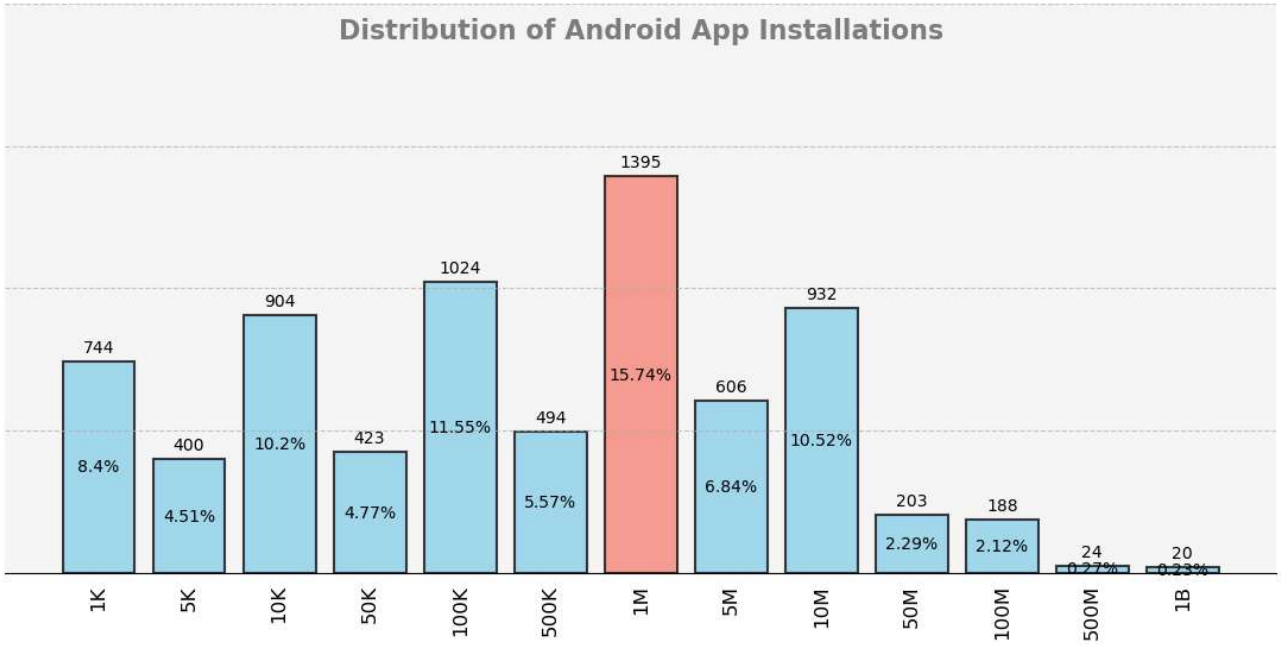
In [32]:
```
install_frq.index= install_frq.index.map(alphanum_units)
install_frq
```

Out[32]:
```
1K       744
5K       400
10K      904
50K      423
100K    1024
500K     494
1M      1395
5M       606
10M      932
50M      203
100M     188
500M      24
1B        20
Name: Installs_int, dtype: int64
```

GENERATING GRAPH FOR DISTRIBUTION OF APPS ACCORDING TO NO OF INSTALLATIONS

In [33]:
```
categories = install_frq.index
counts =install_frq.values
percentage = install_frq_per.values

plt.figure(figsize = (12,7))
bars = plt.bar(categories, counts, color = "skyblue", alpha=0.75, edgecolor= "black", linewidth=1.5)
plt.xticks(rotation=90, fontsize=12)
plt.yticks(fontsize=12)
plt.grid(axis="y",linestyle="--", alpha=0.7)
plt.grid(axis="x",linestyle="")
plt.xticks(fontsize=12)
plt.yticks(range(0,2500,500),[], fontsize=12)
plt.tick_params(bottom = 0, left = 0)
max_count_category= categories[counts.argmax()]
max_count_index= list(categories).index(max_count_category)
bars[max_count_index].set_color("salmon")
bars[max_count_index].set_edgecolor("black")
for bar , prec in zip(bars, percentage):
    height = bar.get_height()
    plt.text(bar.get_x()+bar.get_width()/2, height +20, '%d' % int(height), ha= 'center', va='bottom', fontsize=
    plt.text(bar.get_x()+bar.get_width()/2, height/2, f'{prec}%', ha= 'center', va='center', fontsize=10, color=
ax = plt.gca()
ax.set_facecolor('#f7f7f7')
plt.text(0.5,0.94,'Distribution of Android App Installations', horizontalalignment= 'center', fontsize=16, trans
plt.text(0.5,-.35,'Based on the provided data, its clear that the bulk of Android app installations are concentra
for i in ["top","right","left"]:
    plt.gca().spines[i].set_visible(False)
plt.tight_layout()
plt.savefig('TOP INSTALLATIONS.jpg', format='jpeg')
plt.show()
```



Based on the provided data, its clear that the bulk of Android app installations are concentrated in the lower range, notably within the 1K to 1M installation range. Particularly, the 1M installation range appears prominently with 1395 apps, signifying a substantial portion of apps falling into this category. As the number of installations increases, the count of apps decreases, with only a small number of apps achieving installation counts of 500M and 1B.

In [34]:
```
categories_android= android_final["Category"].unique()
categories_android
```

Out[34]:
```
array(['ART_AND_DESIGN', 'AUTO_AND_VEHICLES', 'BEAUTY',
       'BOOKS_AND_REFERENCE', 'BUSINESS', 'COMICS', 'COMMUNICATION',
       'DATING', 'EDUCATION', 'ENTERTAINMENT', 'EVENTS', 'FINANCE',
       'FOOD_AND_DRINK', 'HEALTH_AND_FITNESS', 'HOUSE_AND_HOME',
       'LIBRARIES_AND_DEMO', 'LIFESTYLE', 'GAME', 'FAMILY', 'MEDICAL',
       'SOCIAL', 'SHOPPING', 'PHOTOGRAPHY', 'SPORTS', 'TRAVEL_AND_LOCAL',
       'TOOLS', 'PERSONALIZATION', 'PRODUCTIVITY', 'PARENTING', 'WEATHER',
       'VIDEO_PLAYERS', 'NEWS_AND_MAGAZINES', 'MAPS_AND_NAVIGATION'],
      dtype=object)
```

AVRAGE DISTRIBUTIONS OF ANDROID APP INSTALLATIONS BY CATEGORIES

In [35]:
```
pd.pivot_table(android_final, values="Installs_int", index= 'Category', aggfunc='mean')
```

Out[35]:

Category	Installs_int
ART_AND_DESIGN	1.986335e+06
AUTO_AND_VEHICLES	6.473178e+05
BEAUTY	5.131519e+05
BOOKS_AND_REFERENCE	8.767812e+06
BUSINESS	1.712290e+06
COMICS	8.176573e+05
COMMUNICATION	3.845612e+07
DATING	8.540288e+05
EDUCATION	1.820673e+06
ENTERTAINMENT	1.164071e+07
EVENTS	2.535422e+05
FAMILY	3.694276e+06
FINANCE	1.387692e+06
FOOD_AND_DRINK	1.924898e+06
GAME	1.556097e+07
HEALTH_AND_FITNESS	4.188822e+06
HOUSE_AND_HOME	1.331541e+06
LIBRARIES_AND_DEMO	6.385037e+05
LIFESTYLE	1.437816e+06
MAPS_AND_NAVIGATION	4.056942e+06
MEDICAL	1.206165e+05
NEWS_AND_MAGAZINES	9.549178e+06
PARENTING	5.426036e+05
PERSONALIZATION	5.201483e+06
PHOTOGRAPHY	1.780563e+07
PRODUCTIVITY	1.678733e+07
SHOPPING	7.036877e+06
SOCIAL	2.325365e+07
SPORTS	3.638640e+06
TOOLS	1.068230e+07
TRAVEL_AND_LOCAL	1.398408e+07
VIDEO_PLAYERS	2.472787e+07
WEATHER	5.074486e+06

In [36]:
```
pd.options.display.float_format= '{:.0f}'.format
```

In [37]:
```
categories_installs= pd.pivot_table(android_final, values="Installs_int", index= 'Category', aggfunc='mean')
categories_installs = categories_installs.sort_values(by= "Installs_int",ascending= False)
categories_installs = categories_installs["Installs_int"]
categories_installs
```

Out[37]:
```
Category
COMMUNICATION         38456119
VIDEO_PLAYERS         24727872
SOCIAL                23253652
PHOTOGRAPHY           17805628
PRODUCTIVITY          16787331
GAME                  15560966
TRAVEL_AND_LOCAL      13984078
ENTERTAINMENT         11640706
TOOLS                 10682301
NEWS_AND_MAGAZINES     9549178
BOOKS_AND_REFERENCE    8767812
SHOPPING               7036877
PERSONALIZATION        5201483
WEATHER                5074486
HEALTH_AND_FITNESS     4188822
MAPS_AND_NAVIGATION    4056942
FAMILY                 3694276
SPORTS                 3638640
ART_AND_DESIGN         1986335
FOOD_AND_DRINK         1924898
EDUCATION              1820673
BUSINESS               1712290
LIFESTYLE              1437816
FINANCE                1387692
HOUSE_AND_HOME         1331541
DATING                  854029
COMICS                  817657
AUTO_AND_VEHICLES       647318
LIBRARIES_AND_DEMO      638504
PARENTING               542604
BEAUTY                  513152
EVENTS                  253542
MEDICAL                 120616
Name: Installs_int, dtype: float64
```

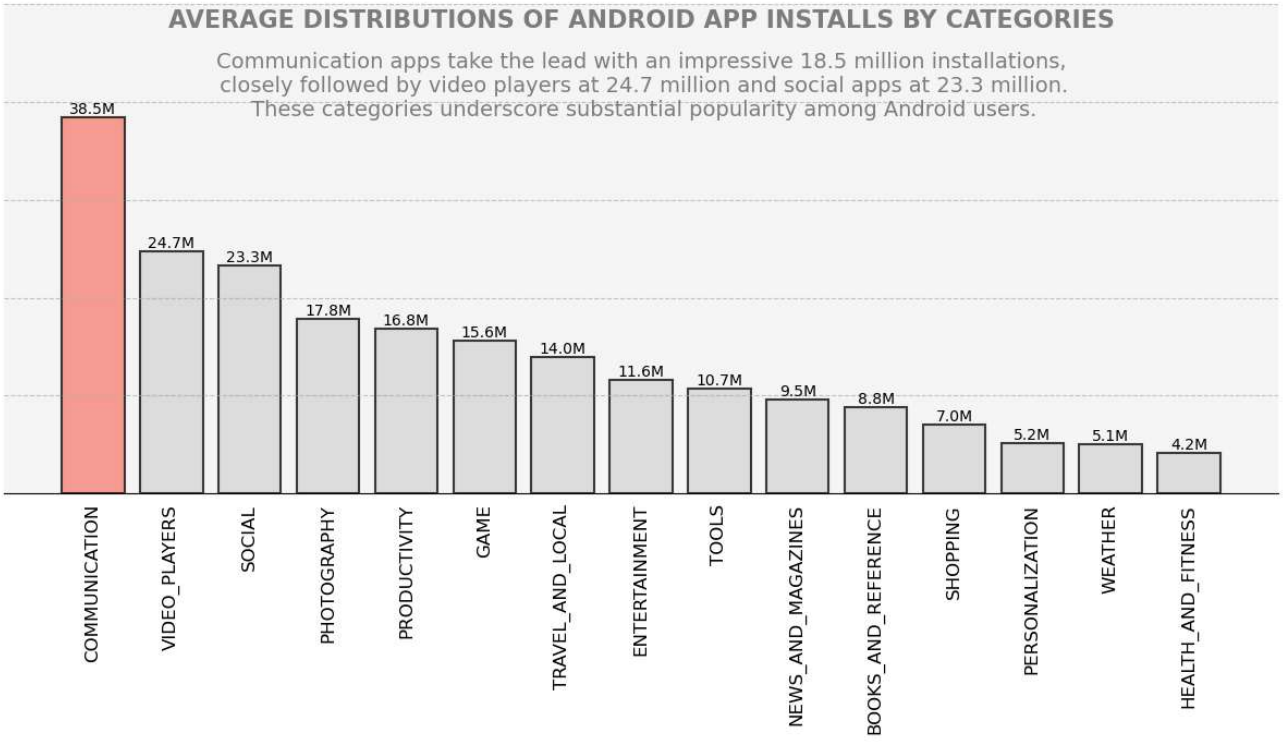
In [38]:
```
def alphanum_units01(value):
    if value >=1e9:
        return f'{value/1e9:.1f}B'
    elif value >=1e6:
        return f'{value/1e6:.1f}M'
    elif value >=1e3:
        return f'{value/1e3:.1f}K'
    else:
        return f'{value:.1f}'
```

In [39]:
```
categories_installs_units = categories_installs.map(alphanum_units01)
categories_installs_units
```

Out[39]:
```
Category
COMMUNICATION           38.5M
VIDEO_PLAYERS           24.7M
SOCIAL                  23.3M
PHOTOGRAPHY             17.8M
PRODUCTIVITY            16.8M
GAME                    15.6M
TRAVEL_AND_LOCAL        14.0M
ENTERTAINMENT           11.6M
TOOLS                   10.7M
NEWS_AND_MAGAZINES       9.5M
BOOKS_AND_REFERENCE      8.8M
SHOPPING                 7.0M
PERSONALIZATION          5.2M
WEATHER                  5.1M
HEALTH_AND_FITNESS       4.2M
MAPS_AND_NAVIGATION      4.1M
FAMILY                   3.7M
SPORTS                   3.6M
ART_AND_DESIGN           2.0M
FOOD_AND_DRINK           1.9M
EDUCATION                1.8M
BUSINESS                 1.7M
LIFESTYLE                1.4M
FINANCE                  1.4M
HOUSE_AND_HOME           1.3M
DATING                 854.0K
COMICS                 817.7K
AUTO_AND_VEHICLES      647.3K
LIBRARIES_AND_DEMO     638.5K
PARENTING              542.6K
BEAUTY                 513.2K
EVENTS                 253.5K
MEDICAL                120.6K
Name: Installs_int, dtype: object
```

In [40]:
```
categories = categories_installs.index[:15]
counts = categories_installs.values[:15]

plt.figure(figsize = (12,7))
bars = plt.bar(categories, counts, color = "lightgray", alpha=0.75, edgecolor= "black", linewidth=1.5)
plt.xticks(rotation=90, fontsize=12)
plt.yticks(fontsize=12)
plt.grid(axis="y",linestyle="--", alpha=0.7)
plt.grid(axis="x",linestyle="")
plt.xticks(fontsize=12)
plt.yticks(range(0,60000000,10000000),[], fontsize=12)
plt.tick_params(bottom = 0, left = 0)
max_count_category= categories[counts.argmax()]
max_count_index= list(categories).index(max_count_category)
bars[max_count_index].set_color("salmon")
bars[max_count_index].set_edgecolor("black")
for bar , units in zip(bars, categories_installs_units.values):
    height = bar.get_height()
    plt.text(bar.get_x()+bar.get_width()/2, height +25, units, ha= 'center', va='bottom', fontsize=10)
ax = plt.gca()
ax.set_facecolor('#f7f7f7')
plt.text(0.5,0.95,'AVERAGE DISTRIBUTIONS OF ANDROID APP INSTALLS BY CATEGORIES', horizontalalignment= 'center',
plt.text(0.5,0.77,'Communication apps take the lead with an impressive 18.5 million installations,\n closely foll
for i in ["top","right","left"]:
    plt.gca().spines[i].set_visible(False)
plt.tight_layout()
plt.savefig('TOP CATEGORIY INSTALLATIONS.jpg', format='jpeg')
plt.show()
```



In [41]:
```
categories_group = android_final.groupby('Category')
```

In [42]:
```
categories_installs.index[:15]
```

Out[42]:
```
Index(['COMMUNICATION', 'VIDEO_PLAYERS', 'SOCIAL', 'PHOTOGRAPHY',
       'PRODUCTIVITY', 'GAME', 'TRAVEL_AND_LOCAL', 'ENTERTAINMENT', 'TOOLS',
       'NEWS_AND_MAGAZINES', 'BOOKS_AND_REFERENCE', 'SHOPPING',
       'PERSONALIZATION', 'WEATHER', 'HEALTH_AND_FITNESS'],
      dtype='object', name='Category')
```

In [43]:
```
df= categories_group.get_group('COMMUNICATION').sort_values(by= 'Installs_int',ascending = False)
df = df[["App","Installs_int"]].head(15)
df["Installs_int_units"] = df["Installs_int"].map(alphanum_units)
df
```

Out[43]:

	App	Installs_int	Installs_int_units
336	WhatsApp Messenger	1000000000	1B
382	Messenger – Text and Video Chat for Free	1000000000	1B
464	Hangouts	1000000000	1B
411	Google Chrome: Fast & Secure	1000000000	1B
391	Skype - free IM & video calls	1000000000	1B
451	Gmail	1000000000	1B
403	LINE: Free Calls & Messages	500000000	500M
4676	Viber Messenger	500000000	500M
420	UC Browser - Fast Download Private & Secure	500000000	500M
371	Google Duo - High Quality Video Calls	500000000	500M
383	imo free video calls and chat	500000000	500M
393	Who	100000000	100M
4633	UC Browser Mini -Tiny Fast Private & Secure	100000000	100M
4602	Truecaller: Caller ID, SMS spam blocking & Dialer	100000000	100M
4592	Telegram	100000000	100M

In [44]:
```
df= categories_group.get_group('SOCIAL').sort_values(by= 'Installs_int',ascending = False)
df = df[["App","Installs_int"]].head(15)
df["Installs_int_units"] = df["Installs_int"].map(alphanum_units)
df
```

Out[44]:

	App	Installs_int	Installs_int_units
2544	Facebook	1000000000	1B
2554	Google+	1000000000	1B
2604	Instagram	1000000000	1B
2610	Snapchat	500000000	500M
2546	Facebook Lite	500000000	500M
3945	Tik Tok - including musical.ly	100000000	100M
2592	Tango - Live Video Broadcast	100000000	100M
6373	VK	100000000	100M
2552	Pinterest	100000000	100M
3951	BIGO LIVE - Live Stream	100000000	100M
2621	LinkedIn	100000000	100M
2548	Tumblr	100000000	100M
2588	Badoo - Free Chat & Dating App	100000000	100M
2636	Zello PTT Walkie Talkie	50000000	50M
2595	ooVoo Video Calls, Messaging & Stories	50000000	50M

In [45]:
```
df= categories_group.get_group('PHOTOGRAPHY').sort_values(by= 'Installs_int',ascending = False)
df = df[["App","Installs_int"]].head(15)
df["Installs_int_units"] = df["Installs_int"].map(alphanum_units)
df
```

Out[45]:

	App	Installs_int	Installs_int_units
2884	Google Photos	1000000000	1B
4574	S Photo Editor - Collage Maker , Photo Collage	100000000	100M
2949	Camera360: Selfie Photo Editor with Funny Sticker	100000000	100M
2908	Retrica	100000000	100M
8307	LINE Camera - Photo editor	100000000	100M
2921	Photo Editor Pro	100000000	100M
2847	Sweet Selfie - selfie camera, beauty cam, phot...	100000000	100M
2937	BeautyPlus - Easy Photo Editor & Selfie Camera	100000000	100M
2938	PicsArt Photo Studio: Collage Maker & Pic Editor	100000000	100M
5057	AR effect	100000000	100M
2833	YouCam Makeup - Magic Selfie Makeovers	100000000	100M
2942	Z Camera - Photo Editor, Beauty Selfie, Collage	100000000	100M
2943	PhotoGrid: Video & Pic Collage Maker, Photo Ed...	100000000	100M
2944	Candy Camera - selfie, beauty camera, photo ed...	100000000	100M
2945	YouCam Perfect - Selfie Photo Editor	100000000	100M

In [46]:
```
df= categories_group.get_group('VIDEO_PLAYERS').sort_values(by= 'Installs_int',ascending = False)
df = df[["App","Installs_int"]].head(15)
df["Installs_int_units"] = df["Installs_int"].map(alphanum_units)
df
```

Out[46]:

	App	Installs_int	Installs_int_units
3665	YouTube	1000000000	1B
3687	Google Play Movies & TV	1000000000	1B
3711	MX Player	500000000	500M
3675	VLC for Android	100000000	100M
4688	VivaVideo - Video Editor & Photo Movie	100000000	100M
4032	Dubsmash	100000000	100M
10647	Motorola FM Radio	100000000	100M
4696	VideoShow-Video Editor, Video Maker, Beauty Ca...	100000000	100M
3672	Motorola Gallery	100000000	100M
3691	Samsung Video Library	50000000	50M
4038	DU Recorder – Screen Recorder, Video Editor, Live	50000000	50M
3693	LIKE – Magic Video Maker & Community	50000000	50M
3686	Vigo Video	50000000	50M
4049	KineMaster – Pro Video Editor	50000000	50M
5612	Ringdroid	50000000	50M

In [47]:
```
df= categories_group.get_group('ENTERTAINMENT').sort_values(by= 'Installs_int',ascending = False)
df = df[["App","Installs_int"]].head(15)
df["Installs_int_units"] = df["Installs_int"].map(alphanum_units)
df
```

Out[47]:

	App	Installs_int	Installs_int_units
874	Talking Angela	100000000	100M
866	Hotstar	100000000	100M
888	IMDb Movies & TV	100000000	100M
958	Netflix	100000000	100M
893	Talking Ben the Dog	100000000	100M
879	Talking Ginger 2	50000000	50M
889	Twitch: Livestream Multiplayer Games & Esports	50000000	50M
886	Amazon Prime Video	50000000	50M
892	PlayStation App	50000000	50M
863	Motorola Spotlight Player™	10000000	10M
860	Mobile TV	10000000	10M
973	Redbox	10000000	10M
885	BBC Media Player	10000000	10M
971	Viki: Asian TV Dramas & Movies	10000000	10M
883	Movies by Flixster, with Rotten Tomatoes	10000000	10M

In [48]:
```
df= categories_group.get_group('TRAVEL_AND_LOCAL').sort_values(by= 'Installs_int',ascending = False)
df = df[["App","Installs_int"]].head(15)
df["Installs_int_units"] = df["Installs_int"].map(alphanum_units)
df
```

Out[48]:

	App	Installs_int	Installs_int_units
3223	Maps - Navigate & Explore	1000000000	1B
3232	Google Street View	1000000000	1B
3136	TripAdvisor Hotels Flights Restaurants Attract...	100000000	100M
3112	Booking.com Travel Deals	100000000	100M
9841	Google Earth	100000000	100M
3151	2GIS: directory & navigator	50000000	50M
3125	VZ Navigator	50000000	50M
3103	trivago: Hotels & Travel	50000000	50M
9833	MAPS.ME — Offline Map and Travel Navigation	50000000	50M
3217	Agoda — Hotel Booking Deals	10000000	10M
3163	MakeMyTrip-Flight Hotel Bus Cab IRCTC Rail Boo...	10000000	10M
3162	Where is my Train : Indian Railway & PNR Status	10000000	10M
3215	Airbnb	10000000	10M
3224	GPS Status & Toolbox	10000000	10M
3169	Yelp: Food, Shopping, Services Nearby	10000000	10M

In [49]:
```
df= categories_group.get_group('TOOLS').sort_values(by= 'Installs_int',ascending = False)
df = df[["App","Installs_int"]].head(15)
df["Installs_int_units"] = df["Installs_int"].map(alphanum_units)
df
```

Out[49]:

	App	Installs_int	Installs_int_units
3234	Google	1000000000	1B
3265	Gboard - the Google Keyboard	500000000	500M
3255	SHAREit - Transfer & Share	500000000	500M
4005	Clean Master- Space Cleaner & Antivirus	500000000	500M
3235	Google Translate	500000000	500M
7536	Security Master - Antivirus, VPN, AppLock, Boo...	500000000	500M
8452	Automatic Call Recorder	100000000	100M
3266	Google Korean Input	100000000	100M
7550	Battery Doctor-Battery Life Saver & Battery Co...	100000000	100M
3272	Share Music & Transfer Files - Xender	100000000	100M
4578	Samsung Smart Switch Mobile	100000000	100M
4568	360 Security - Free Antivirus, Booster, Cleaner	100000000	100M
3289	Tiny Flashlight + LED	100000000	100M
4151	Google Now Launcher	100000000	100M
8758	Anti-virus Dr.Web Light	100000000	100M

CONCLUSION

IN CONCLUSION, IT CAN BE INFERRED THAT THERE ARE AMPLE OPPORTUNITIES FOR IMPROVEMENT ACROSS VARIOUS APP CATEGORIES. WHILE CERTAIN CATEGORIES DOMINATED BY TECH GIANTS LIKE GOOGLE, META, AND MICROSOFT-BACKED APPS HOLD TOP POSITIONS, THERE REMAINS SIGNIFICANT ROOM FOR EMERGING GIANTS TO LEVERAGE ARTIFICIAL INTELLIGENCE TOOLS IN THEIR APPS. AS EVIDENCED BY RECENT ACHIEVEMENTS SUCH AS CHATGPT BREAKING RECORDS WITH 100 MILLION MONTHLY ACTIVE USERS JUST TWO MONTHS AFTER LAUNCH, THERE IS POTENTIAL FOR RAPID GROWTH AND INNOVATION WITHIN THE INDUSTRY.

In [ ]:
```
```