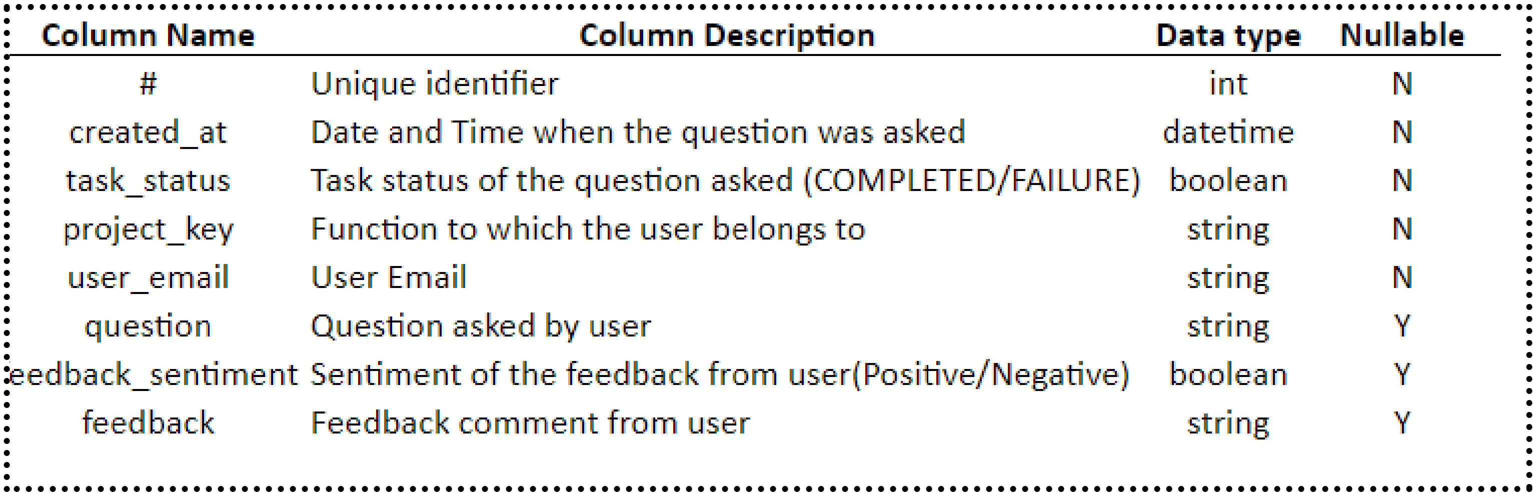
User Segmentation and Retention Analysis for AI Chatbot

Overview: The data belongs to a Gen AI-based product designed to provide contextual insights based on structured and unstructured databases. This product answers user queries in natural language. Within seconds, Insights Copilot not only summarizes responses but also generates and visualizes valuable insights.

Data Dictionary -



Column Name	Column Description	Data type	Nullable
#	Unique identifier	int	N
created_at	Date and Time when the question was asked	datetime	N
task_status	Task status of the question asked (COMPLETED/FAILURE)	boolean	N
project_key	Function to which the user belongs to	string	N
user_email	User Email	string	N
question	Question asked by user	string	Y
eedback_sentiment	Sentiment of the feedback from user(Positive/Negative)	boolean	Y
feedback	Feedback comment from user	string	Y

Identifying metrics

User experience - highly impactful. Users should easily find answers to their questions.
To measure this

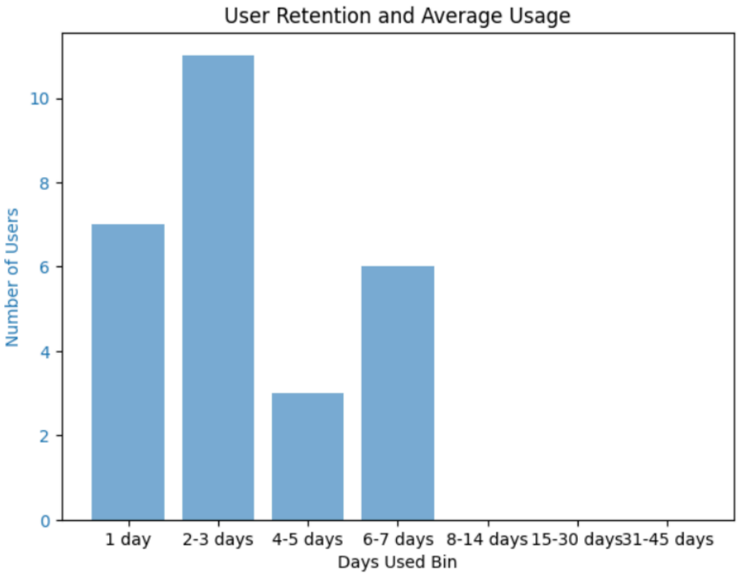
- User feedback - Positive
- User retention - transaction frequency
- Engagement
- Activation rate - A strong onboarding process is crucial for creating a positive first impression and encouraging user acquisition.
-

Data specifics

Total records - 839
Total unique users - 27
The data was from 4/6/24 to 18/7/24 - 45 days but only 29 unique dates present

My goal was to segment users into cohorts based on their behavior with the product. After analyzing the data, I observed that some users had peak usage, visiting the site 5-10 times a day, while others did not return for many days.

To start, I grouped the data by `user_email` and calculated the number of unique dates each user visited the site. This allowed me to determine how many days each user revisited the site. I then categorized these days into buckets and plotted the number of users falling into each bucket.

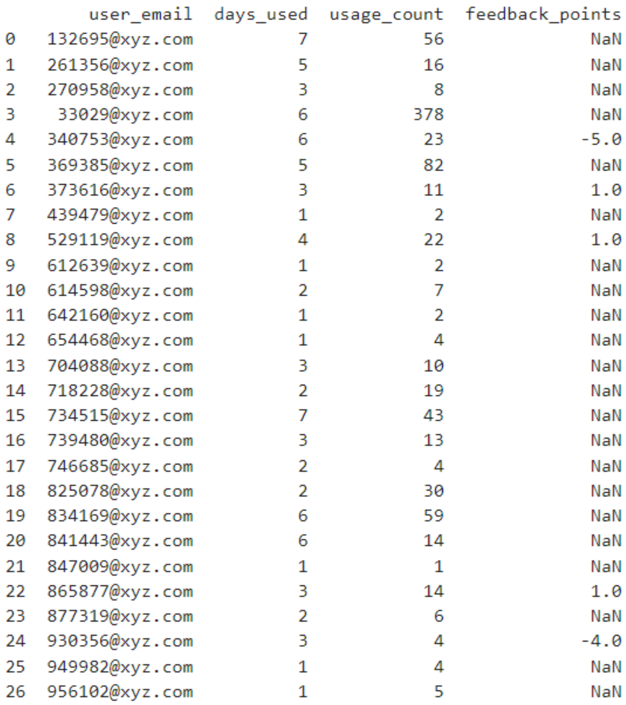


I then categorized the number of days into buckets and plotted the number of users in each bucket. From this, it was clear that users in the 6-7 day bucket were consistently returning to the site, while users with only 1 day of activity might need activation

However, distinguishing between users in the 2-3 day bucket and the 4-5 day bucket was challenging. To address this, I calculated the number of queries each user made and joined this information with the number of days they returned to the app.

It's possible that a user with high usage might have frequently queried without finding satisfactory answers. Therefore, I wanted to incorporate user feedback into the analysis.

To merge user feedback with the usage data, I first calculated feedback points.



```
        user_email  days_used  usage_count  feedback_points
0   132695@xyz.com          7           56              NaN
1   261356@xyz.com          5           16              NaN
2   270958@xyz.com          3            8              NaN
3    33029@xyz.com          6          378              NaN
4   340753@xyz.com          6           23             -5.0
5   369385@xyz.com          5           82              NaN
6   373616@xyz.com          3           11              1.0
7   439479@xyz.com          1            2              NaN
8   529119@xyz.com          4           22              1.0
9   612639@xyz.com          1            2              NaN
10  614598@xyz.com          2            7              NaN
11  642160@xyz.com          1            2              NaN
12  654468@xyz.com          1            4              NaN
13  704088@xyz.com          3           10              NaN
14  718228@xyz.com          2           19              NaN
15  734515@xyz.com          7           43              NaN
16  739480@xyz.com          3           13              NaN
17  746685@xyz.com          2            4              NaN
18  825078@xyz.com          2           30              NaN
19  834169@xyz.com          6           59              NaN
20  841443@xyz.com          6           14              NaN
21  847009@xyz.com          1            1              NaN
22  865877@xyz.com          3           14              1.0
23  877319@xyz.com          2            6              NaN
24  930356@xyz.com          3            4             -4.0
25  949982@xyz.com          1            4              NaN
26  956102@xyz.com          1            5              NaN
```

Each positive feedback sentiment was assigned a +1 point, while each negative feedback was assigned a -1 point. This allowed me to calculate a feedback score for each user.

Given that the number of users providing feedback was relatively low, there were many null values. To address this, I performed a left join with the existing usage data table.

After merging, I applied conditions to classify users, as

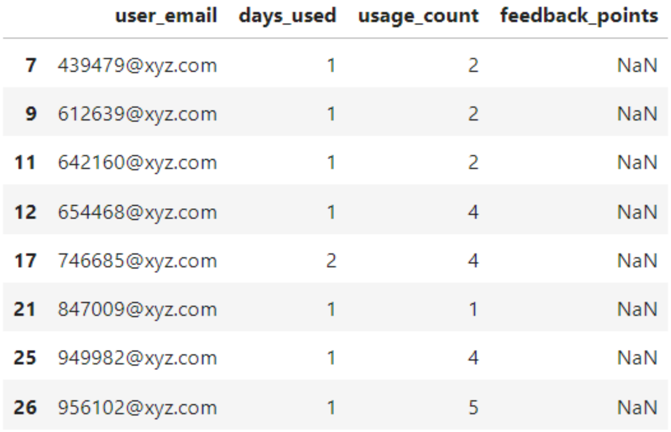
- Power users
- Casual Users
- Users Needing Activation
- Unsatisfied Users

Users Needing Activation

Conditions:

- Days used < 3
- Usage count < 6
- Feedback points: No conditions applied (assuming these users are not fully introduced to the *app)*

(These criteria are considered as activation metrics.)



	user_email	days_used	usage_count	feedback_points
7	439479@xyz.com	1	2	NaN
9	612639@xyz.com	1	2	NaN
11	642160@xyz.com	1	2	NaN
12	654468@xyz.com	1	4	NaN
17	746685@xyz.com	2	4	NaN
21	847009@xyz.com	1	1	NaN
25	949982@xyz.com	1	4	NaN
26	956102@xyz.com	1	5	NaN

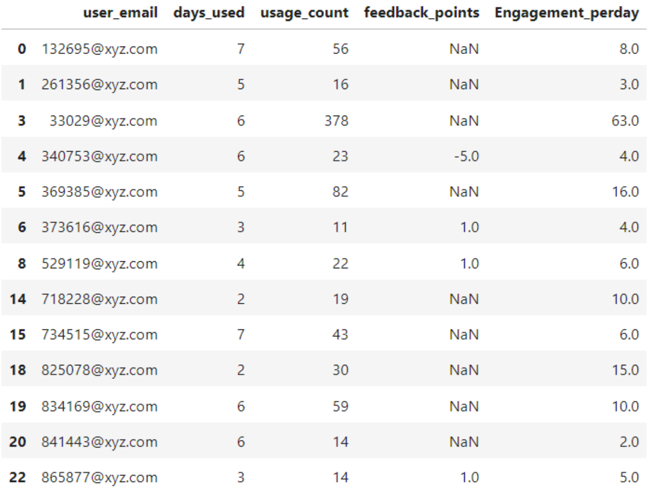
Power Users Cohort

Conditions:

- Days used ≥ 4
- Usage count > 30
- Feedback points ≥ 0
- Engagement per day ≥ 10

Engagement Quotient:

- Calculated as the number of queries divided by the number of days questions were asked (like , average session time, average number of engaging questions)

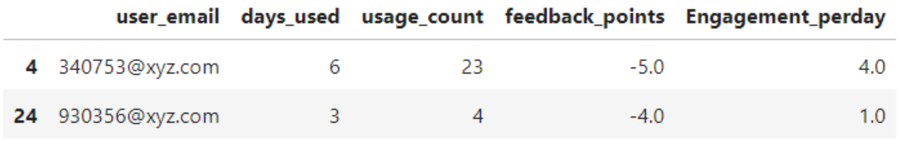


	user_email	days_used	usage_count	feedback_points	Engagement_perday
0	132695@xyz.com	7	56	NaN	8.0
1	261356@xyz.com	5	16	NaN	3.0
3	33029@xyz.com	6	378	NaN	63.0
4	340753@xyz.com	6	23	-5.0	4.0
5	369385@xyz.com	5	82	NaN	16.0
6	373616@xyz.com	3	11	1.0	4.0
8	529119@xyz.com	4	22	1.0	6.0
14	718228@xyz.com	2	19	NaN	10.0
15	734515@xyz.com	7	43	NaN	6.0
18	825078@xyz.com	2	30	NaN	15.0
19	834169@xyz.com	6	59	NaN	10.0
20	841443@xyz.com	6	14	NaN	2.0
22	865877@xyz.com	3	14	1.0	5.0

Unsatisfied Users

Conditions:

- Feedback points < 0



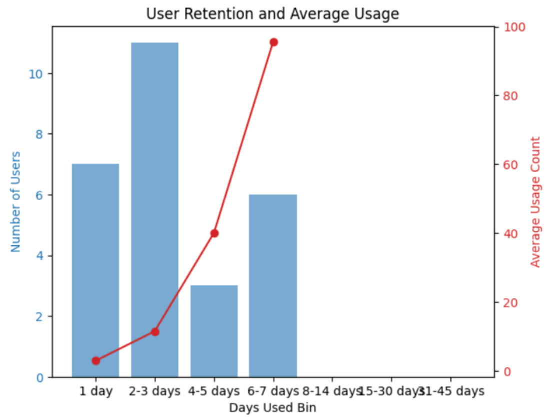
	user_email	days_used	usage_count	feedback_points	Engagement_perday
4	340753@xyz.com	6	23	-5.0	4.0
24	930356@xyz.com	3	4	-4.0	1.0

We can collect personal feedback from h users with -ve feedback to gain a better understanding of what's lacking.

Casual Users
The remaining users who do not fall into other categories **(**These users have completed activation and use the website at a moderate level.)

Retention Indicators -

- (no.of days a user returned)
- User Engagement Metrics (avg engagement_perday)
- Customer Feedback (feedback_points)



Red line - the average number of queries
Bar graph - number of users falling into different buckets based on the number of days they accessed the site (e.g., days 1-2, 3-4)
From the graph, it's clear that users who visited the site for more than 4-5 days are more active and likely to return,
If we consider the user segments, the power users and some casual users are most likely to return. However, the retention of users needing activation and unsatisfied users remains doubtful.

Gaps in Current Usage Data

1. Precision of Engagement Time:- The current tools do not provide precise measurement of user engagement time
2. Data Extraction Difficulty: Extracting meaningful insights from raw data is complex
3. Ease of Funnel and Cohort Creation: Despite the abundance of data, the current tool excels at creating funnels and cohorts with multiple filters, allowing for detailed segmentation and analysis of user behavior.
4. Accurate Live Tracking: They provide precise live tracking of user engagement times.

Strengths and Weaknesses of Analytics Tools

1. Amplitude:-

- Detailed and precise real-time user behavior tracking but can be complex to set up and may require significant customization.

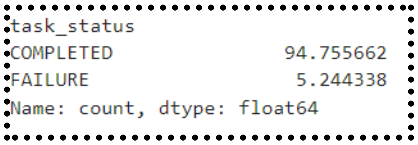
2. Mixpanel :-

- Excellent real-time engagement metrics with an intuitive interface for funnel and cohort analysis but higher cost compared to some alternatives, especially at scale.

Success Criteria for the Chat Bar Feature
High Usage Rate: The chat bar should be used frequently by a significant portion of users.
User Satisfaction: Users should find the chat bar feature helpful and easy to use.
Effective Query Resolution: The feature should effectively address user queries and provide valuable insights.

Metrics to Track
Query effectiveness:

- Query Resolution Rate: Percentage of queries that are successfully answered by the chat bar.
- Response Accuracy: User ratings or feedback on the relevance and accuracy of responses provided by the chat bar.

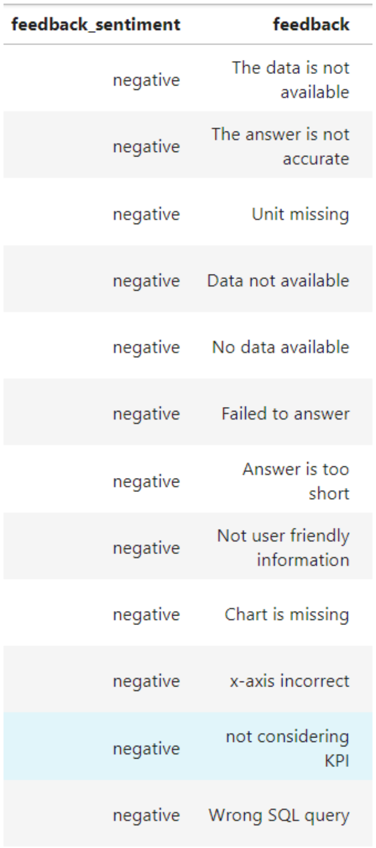


```
task_status
COMPLETED          94.755662
FAILURE             5.244338
Name: count, dtype: float64
```

Engagement metrics :

Usage frequency : how often users interact with the chat bar.

User feedback:



feedback_sentiment	feedback
negative	The data is not available
negative	The answer is not accurate
negative	Unit missing
negative	Data not available
negative	No data available
negative	Failed to answer
negative	Answer is too short
negative	Not user friendly information
negative	Chart is missing
negative	x-axis incorrect
negative	not considering KPI
negative	Wrong SQL query

Based on user feedback and data points, the response accuracy of the chat bar needs improvement. Users are dissatisfied with the data, charts, and details provided. Additionally, feedback indicates that some details are missing.

The feedback and data currently available are insufficient to clearly identify user difficulties. To address this, we should simplify the feedback mechanism to encourage more user input. Increased feedback can help us enhance the feature. Additionally, by gathering more data points, we can better analyze where users are encountering issues and make informed improvements in chat bar

LINK TO THE CODE - CLICK HERE