Data Management for Data Science Final Project

December 2024

By Alexander Mandryk and Avi Herskowitz (Rutgers University)

Defining the Project

What problem are we solving? This project focuses on analyzing customer sentiment from the chat logs of TEMO Sunrooms, a leading U.S. manufacturer specializing in high-quality sunrooms, pergolas, patio covers, and other outdoor living solutions, offering custom-built, durable, and low-maintenance products backed by a limited lifetime warranty. The dataset contains approximately 2,000 detailed chat records, including aspects such as unique chat IDs, timestamps, user location, referral links, lead types (categorical: None [0], Sales [1], Service [2], and Other [3]), transcript text, and visitor IDs.

The key problem we aim to solve is identifying trends in customer sentiment across variables presently available in the data through analysis and feature engineering. By analyzing sentiment trends, we intend to provide actionable insights to TEMO Sunrooms to optimize their customer service strategies, identify potential areas for operational improvement, enhance customer satisfaction, and drive overall revenue growth.

What Strategic Aspects are involved? The strategic aspects involved with our project have four main focus areas - improving the customer experience, optimizing processes within TEMO, providing actionable insights to allow for more targeted marketing and sales outreach, and possibly uncovering an ideal customer description. To achieve these aspects we begin by identifying gaps within data to engineer numerical features for analysis, we then breakdown the text to analyze the sentiment of every chat, and finally, we analyze sentiment patterns—with respect to lead type, referrer, chat length, user geolocation data, and many other aspects related to the user and their chat—to help target marketing and sales teams within TEMO.

Relation to Course Lectures and Research**:** This project reflects concepts discussed in class and recitation, such as:

- *Data Cleaning and Preprocessing***:** Techniques like handling missing data, tokenization, one-hot-encoding, stop word removal, feature engineering, and RegEx pattern identification, were emphasized in lectures.
    - Heavy emphasis on creating numerical data in order to maximize the data analyzed through our K-Means Clustering implementation
- *Sentiment Analysis:* We used the VADER sentiment analysis tool to derive polarity scores from text, an application of natural language processing (NLP) concepts.
- *Graphical Representation:* Once we retrieved the data from the chat logs we created graphical representations—using seaborn—as discussed in class to illustrate our finding to the user.
- *Machine Learning Applications:* Implemented K-Means Clustering to isolate groups within the data to further analyze for trends. This was completed with scaled data in order to allow for appropriate, balanced analysis on 10+ numerical feature points.
- *Database Management:* Post-cleaning, and analysis the data was structured and prepared for SQL database integration, for easy retrieval by the user. 6 unique tables were stored within the SQLAlchemy engine; each table sharing a Chat ID column to allow for joining, grouping, and other SQL commands. This data was illustrated prior to K-Means Clustering implementation in order to identify possible trends relating to compounded sentiment score

Importance of the Project

Why is this project important? The project is important because customer sentiment is a critical indicator of the success of any business, this is particularly important for national companies such as TEMO Sunrooms that must provide personalized yet reproducible service that allows them to connect with customers located throughout the United States. We hope that by analyzing these interactions we can reveal patterns that will help guide TEMO Sunrooms executives in their decision making processes. We were personally tasked with providing insights that enable the C-suite at TEMO Sunrooms to execute informed decisions that drive revenue growth and expand TEMO Sunrooms' reach whilst maintaining a high quality of service.

Why are we excited about it? We were excited to work on this project as it integrated data of a real world company and included real interactions from customers. This was new to us as prior to this project a lot of the data we were working on seemed more theoretical and was within the defined boundaries of the course work. This project allowed us to branch out to the industry, and realize the potential and power that data analysis can provide on real world companies.

Existing Issues in Data Management Practices: Naturally because we used real world data this came with some pitfalls. Specifically, the data included text from chat logs that was somewhat unstructured, and messy. This made it difficult to analyze the data. However, utilizing the techniques we studied in class allowed us to clean the data deliberately and ultimately enabled our progression and insights. Our data was mostly categorical data with a gold mine of information that, when processed and transformed into numerical data, provides great insight into TEMO Sunrooms website visitors.

Related Works: Before getting started with our project we analyzed the different tools and libraries available for data sentiment analysis. Our project built upon specific libraries such as the Natural Language Toolkit library which provided Velance Aware Dictionary and sEntiment Reasoner (VADER) as a resource for our project.

Data and Techniques

The Data:

The dataset we analyzed was retrieved through the work of group member Alexander Mandryk who worked with TEMO Sunrooms whilst at Princeton Internet Marketing to design and train a Conversation AI Chatbot for TEMO Sunrooms and their dealers. Through TEMO Sunrooms US Director of Sales, Robert Randall, we were given exclusive access to the data in order to conduct our analysis and provide insights applicable to their business decisions.

The dataset is a raw compilation of chat logs—sprinkled with geolocation, timeseries, transcript, and other informative data—for a human-monitored chat that was utilized from Q1 through Q3 of 2024. Approximately 2,000 chat logs are stored within the provided dataset with the following attributes: Unique Chat ID, Time (start date/time and end date/time of chat), User Location Data,

Referral Link, Lead Type (None [0], Sales [1], Service [2], and Other [3]), Transcript of Chat, and Unique Visitor ID. An example of the raw data is provided below in Figure 1:

Chat Id	Created On	Ended On	Location	Referrer	Lead Id	Lead Type Id	Lead Type Name	Original Referrer	Landing Referrer	Transcript Text	Visitor Id
6.5E+07	1/1/2024 2:	1/3/2024 9:	Unknown Ci	https://www.temc	0	0		https://www.goog	https://www.goog	[1/1/2024 5:15:28 PM] Gerard: lt;bgt;Hello. Welcome to TEMO Sunroo	1.5E+09
6.5E+07	1/1/2024 11	1/1/2024 11	Little Elm, TX	https://www.temc	1.6E+07	3	Other			[1/2/2024 2:28:07 AM] Gerard: lt;bgt;Hello. Welcome to TEMO Sunroo	1.5E+09
6.5E+07	1/1/2024 11	1/1/2024 11	Little Elm, TX	https://www.temc	0	0		https://www.temc	https://www.temc	[1/2/2024 2:32:00 AM] Crystal: Thanks for visiting our website. Is there	1.5E+09
6.5E+07	1/2/2024 2:	1/2/2024 2:	Unknown Ci	https://www.temc	1.6E+07	3	Other	https://www.temc	https://www.temc	[1/2/2024 5:11:32 AM] Crystal: lt;bgt;Hello. Welcome to TEMO Sunroo	1.5E+09
6.5E+07	1/2/2024 4:	1/2/2024 4:	Unknown Ci	https://www.temc	1.6E+07	3	Other	https://www.goog	https://www.goog	[1/2/2024 7:47:30 AM] Madelyn: <b><font color="#02B7E0">Live Perso	1.5E+09
6.5E+07	1/2/2024 10	1/4/2024 9:	Tampa, FL, (	https://www.temc	0	0				[1/2/2024 1:47:41 PM] Jarell: <b>Hello. Welcome to TEMO Sunrooms! I	1.5E+09
6.5E+07	1/2/2024 12	1/4/2024 9:	Brandywine,	https://www.temc	0	0		https://www.bing	https://www.bing	[1/2/2024 3:24:57 PM] Jarell: lt;bgt;Hello. Welcome to TEMO Sunroom	1.5E+09
6.5E+07	1/2/2024 1:	1/2/2024 2:	Weatherfor	https://www.temc	1.6E+07	2	Service	https://www.goog	https://www.goog	[1/2/2024 4:40:35 PM] Gerard: lt;bgt;lt;font color=#02B7E0gt;Live Pers	1.5E+09
6.5E+07	1/2/2024 2:	1/2/2024 2:	Unknown Ci	https://www.temc	1.6E+07	3	Other	https://www.goog	https://www.goog	[1/2/2024 5:26:19 PM] Gerard: lt;bgt;lt;font color=#02B7E0gt;Live Pers	1.5E+09
6.5E+07	1/2/2024 4:	1/4/2024 9:	Unknown Ci	https://www.temc	0	0		https://www.goog	https://www.goog	[1/2/2024 7:12:20 PM] Jarell: lt;bgt;lt;font color=#02B7E0gt;Live Person	1.5E+09
6.5E+07	1/2/2024 7:	1/4/2024 9:	Unknown Ci	https://www.temc	0	0		https://www.temc	https://www.temc	[1/2/2024 10:27:55 PM] Madelyn: lt;bgt;lt;font color=#02B7E0gt;Live P	1.5E+09
6.5E+07	1/2/2024 7:	1/2/2024 7:	Weatherfor	https://www.temc	1.6E+07	2	Service			[1/2/2024 10:42:18 PM] Crystal: lt;bgt;lt;font color=#02B7E0gt;Live Per	1.5E+09
6.5E+07	1/2/2024 10	1/2/2024 10	Unknown Ci	https://www.temc	1.6E+07	3	Other	https://www.goog	https://www.goog	[1/3/2024 1:17:26 AM] Jarell: lt;bgt;lt;font color=#02B7E0gt;Live Person	1.5E+09
6.5E+07	1/2/2024 10	1/2/2024 10	Unknown Ci	https://www.temc	1.6E+07	3	Other	https://www.temc	https://www.temc	[1/3/2024 1:35:10 AM] Crystal: lt;bgt;Hello. Welcome to TEMO Sunroo	1.5E+09
6.5E+07	1/3/2024 12	1/3/2024 12	Bothell, WA	https://www.temc	1.6E+07	1	Sales	https://www.temc	https://www.temc	[1/3/2024 3:32:20 AM] Kylie: lt;bgt;Hello. Welcome to TEMO Sunrooms	1.5E+09
6.5E+07	1/3/2024 1:	1/5/2024 9:	Fairborn, OH	https://www.temc	0	0		https://www.temc	https://www.temc	[1/3/2024 4:33:17 AM] Madelyn: lt;bgt;lt;font color=#02B7E0gt;Live Pe	1.5E+09
6.5E+07	1/3/2024 11	1/3/2024 11	Anderson, IN	https://www.temc	1.6E+07	3	Other	https://www.goog	https://www.goog	[1/3/2024 2:14:09 PM] Jarell: lt;bgt;Hello. Welcome to TEMO Sunroom	1.5E+09
6.5E+07	1/3/2024 2:	1/3/2024 2:	Unknown Ci	https://www.temc	0	0		https://www.goog	https://www.goog	[1/3/2024 5:25:56 PM] Jarell: lt;bgt;lt;font color=#02B7E0gt;Live Person	1.5E+09
6.5E+07	1/3/2024 3:	1/3/2024 4:	Unknown Ci	https://www.temc	1.6E+07	1	Sales	https://search.yah	https://search.yah	[1/3/2024 6:58:35 PM] Gerard: lt;bgt;Hello. Welcome to TEMO Sunroo	1.5E+09
6.5E+07	1/3/2024 5:	1/5/2024 9:	San Diego, C	https://www.temc	0	0		https://www.goog	https://www.goog	[1/3/2024 8:21:42 PM] Crystal: lt;bgt;Hello. Welcome to TEMO Sunroo	1.5E+09

Figure 1: Raw Dataset Snippet

Techniques Used:

Data Science Component

Data Cleaning: The dataset we worked on was raw and incomplete which required us to use significant cleaning techniques to be able to use the data to gain insights for TEMO.

Eliminated Redundancy: Redundant information, repeated columns containing company information (all defaulted to TEMO Sunrooms) and the 'Landing Referrer' (less valuable than 'Referrer'), were removed from the data.
Handled Missing Data: Implemented strategies with the intention of completing and maintaining as many rows (chats) as possible. Achieved through imputing most common values found through grouping chats by features and identifiers. Ultimately removed only 4 chats from the dataset as they contained no transcript capable of running sentiment analysis.
Split Location Data: Used Regex to split user location data into City, State, and Country; simultaneously, cleaned all strings and implemented default 'unknown' values for missing records ('Unknown City' for City; 'UNK' for State and Country).
Process Transcripts: Used Regex to clean transcripts (removed timestamps, scrap HTML code, and extra whitespace) and extract personal information inputted by users. Information scraped from chats was used to create a one-hot-encoded set of columns detailing whether a user inputted a name, email address, zip code, or phone number.
NLP: In order to implement the VADER analysis model on the chat logs, we used NLP techniques including tokenization, stemming, and stop-word removal to prepare the data for sentiment analysis.

Feature Engineering: In order to prepare our data for the Machine Learning component we needed to create more meaningful columns.

Chat Duration: Added a column named 'Chat Duration' that stored the number of minutes that the chat lasted, calculated through the difference from the start and end timestamps of each chat session
Engaged User: Added a column named 'Engaged' that stores a 1 if user engaged with the representative during the chat session and a 0 otherwise.
Inactive User: Added a column named 'Inactive' that stores a 1 if chat session was closed due to inactivity and a 0 if the representative manually closed chat.
User Information: Created a table, later stored in SQL database, with one-hot-encoded columns corresponding to user information present in chatlog (name, email address, phone, zip code). This provided insight into how cooperative the user was.

Step 4: Fill in na for 'Landing Referrer' and 'Referrer'

```
#find the % of rows in 'Landing Referrer' that are na
print(f"{len(df_proc[df_proc['Landing Referrer'].isna()])/len(df_proc)*100:.2f}% of rows in 'Landing Referrer' are na.")

#find the % of visitors that have no landing referrer that were inactive
avg_nr_inactive = df_proc[df_proc['Landing Referrer'].isna()]['visitor_engaged'].mean()
print(f'{avg_nr_inactive*100:.2f}% of visitors with no landing referrer were engaged in chat')
    # ~80%
    #this is a large portion of active visitors therefore it would be helpful to fill these na values rather than delete them



#create dataframe, without rows that have na in 'Landing Referrer'
df_lr = df_proc.copy()
df_lr = df_lr[df_lr['Landing Referrer'].notna()]

#group by different referrers to see if the mode is significant enough to be used to fill in the rows that are na
referrers = df_lr.groupby('Landing Referrer').size()
print(f'\nThe largest referrer ({referrers.idxmax()}) covers {referrers.max()/len(df_lr)*100:.2f}% of all non-na rows.')
print(f'The second-largest referrer ({  referrers.nlargest(2).index[1]  }) covers {referrers.nlargest(2).iloc[1] / len(df_lr) * 100:.2f}% of all non-na row
#google home page accounts for 40% of the referrers
    #second place (temo home page) accounts for less than 10%
#this sizable portion of the data means that we can fill na rows with google's home page
```

Figure 2: Code Snippet from Data Cleaning

Machine Learning Component Part 1 - Sentiment Analysis

Sentiment Analysis: We implemented VADER, a prebuilt sentiment analysis model, to analyze chat transcripts and derive compounded sentiment scores for each interaction.

VADER: Implemented VADER on the NLP-processed transcripts to create a dataframe of positive, neutral, negative, and compounded sentiment scores for each chat session.
Dataframe Concatenation: Appended the sentiment analysis records to the master dataframe in order to hold a central dataset with all processed information. This was later broken down into unique tables for SQL database engine.

Step 2: Tokenize and Stem Transcripts.

```
[ ]  def tokenize_transcript(transcript):
       # tokenize data into words
       tokens = word_tokenize(transcript)
       return tokens

     def stem_transcript(tokens):
       stemmed_tokens = [stemmer.stem(token) for token in tokens]
       return " ".join(stemmed_tokens)
```

Step 3: Run VADER Sentiment Analysis

```
# function to analyze sentiment
def analyze_transcript_sentiment(transcript):
    cleaned_text = clean_transcript(transcript)
    tokenized_text = tokenize_transcript(cleaned_text)
    stemmed_text = stem_transcript(tokenized_text)
    sentiment = sia.polarity_scores(stemmed_text)
    return sentiment
```

Figure 3: Code Snippet Showing Preparation for VADER Sentiment Analysis

Database Component:

SQL Integration: After processing our data and retrieving VADER compounded sentiment analysis scores for each chat session, we divided our plethora of data into related, unique tables. These tables were stored within an SQLAlchemy database engine and used to perform various SQL queries. We created the following tables (each had the 'Chat ID' in order to perform queries):

df_lead: Contained information about visitor referrer and referral link (the website that the user held the chat session on along with the path to get there), unique Visitor ID (browser cookie tracking to identify returning visitors), and lead type (described above). Additional column 'temo_landing_referrer' created as 1/0 one-hot whether the user's referrer was a TEMO Sunrooms site or not.
df_customer_info_one_hot: one-hot-encoded breakdown of what personal information was inputted by the user during the chat session (name, phone, email, zip code, city, state).
df_sentiment: sentiment analysis information (positive, neutral, negative, compounded scores)
df_location: geolocation information (city, state, country)
df_chat: chat logistical information (chat duration, active/inactive user, transcript)
df_time: date/time of chat (day of the week, month, during or outside of normal work hours [08:00-18:00 M-F]). Day of the week and month turned numerical with ordinal representation to aid visualization; normal work hours column was one-hot-encoded.

Queries: With an SQL database engine containing 6 unique tables, we executed several join and group queries in order to test sentiment compound score versus our data features. *Example Queries:* Sentiment vs. Day of the week; Sentiment vs. Chat Duration in minutes; Sentiment vs. User-inputted Information; Sentiment vs. Country of User;

Sentiment vs. State of User; Sentiment vs. Lead Type, and many more found in our notebook.

```
#create a SQL Alchemy engine
engine = create_engine('sqlite:////home/apm204/cs210/Final Project/chat_logs.db')

#store each dataframe as a SQL table
df_lead.to_sql('df_lead', engine, index=False, if_exists='replace')
df_customer_info_one_hot.to_sql('df_customer_info_one_hot', engine, index=False, if_exists='replace')
df_sentiment.to_sql('df_sentiment', engine, index=False, if_exists='replace')
df_location.to_sql('df_location', engine, index=False, if_exists='replace')
df_chat.to_sql('df_chat', engine, index=False, if_exists='replace')
df_time.to_sql('df_time', engine, index=False, if_exists='replace')

#verify tables creation
with engine.connect() as conn:
    result = conn.execute(text("SELECT name FROM sqlite_master WHERE type='table';"))
    #print(result.fetchall())

#print('This query will help you determine if sentiment varies by day.')
#This query will help you determine if sentiment varies by day.

query = """
SELECT t.day_of_the_week, AVG(s.compound) as avg_sentiment
FROM df_sentiment s
JOIN df_time t ON s.chat_id = t.chat_id
GROUP BY t.day_of_the_week;
"""
sentiment_day_of_week = pd.read_sql(query, engine)
```

Figure 4: Code Snippet Showing SQL Database Preparation

What was found?

— We noticed that sentiment compound scores tended to be lower for non-TEMO referral sites. TEMO referrers tended to have a compound score around 90% whilst non-TEMO referrers sat around the 80% range.

— Chat sessions where a representative was unable to flag a lead type (Service, Sale, or Other) had an average of 90% compound sentiment whereas flagged lead types averaged at 96%. Further, Sales leads led the way with 98.5% score showing that revenue-focused chats tended to elicit greater sentiment from user and representative. This could be a reflection of company focus or successful targeting/ad campaigns.

— Users chatting during normal working hours generally had greater sentiment at 94.2% whereas those outside of normal working hours averaged 92.9%. This may reflect an increase in spam/illicit chats occurring at random hours throughout the week.

— The most resounding: the more information users provided within transcripts, the greater the compounded sentiment score. Users that shared no personal information had an average sentiment score of 86.7% whereas those who shared their name, email, phone, and zip code enjoyed an average of 98.2% compound score.

Machine Learning Component Part 2 - K-Means Clustering

K-Means Clustering: We implemented K-Means clustering in order to group chat sessions by the characteristics that we engineered and transformed into numerical data. After creating unique clusters, we analyzed trends within the clusters.

> *Standardization:* Prior to implementing K-Means clustering, we standardized all features used for analysis in order to remove bias that may alter the power behind data points.
> *Clustering*: After several iterations, we chose to create 6 clusters as it gave a blend of groups trademarked by their terrible, good, or great compound sentiment score. Our clusters are visualized below in Figure 5 with the following legend:

Cluster 0 — 90.2%	Cluster 1 — 46.8%	Cluster 2 — 96.4%
Cluster 3 — 97.0%	Cluster 4 — 90.3%	Cluster 5 — 97.2%

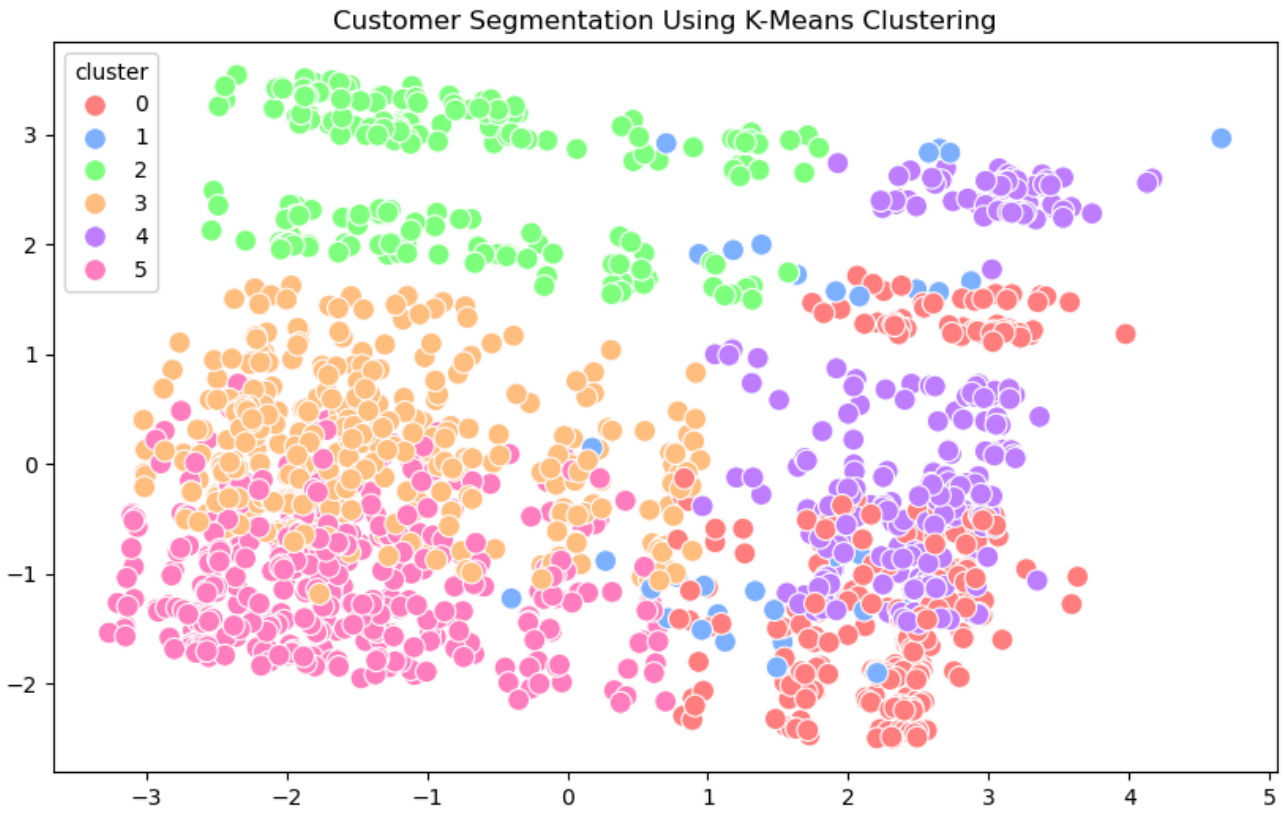

Figure 5: Customer Segmentation using K-Means Clustering

Information Visualization / Analysis Component

Cluster Analysis: After creating unique clusters with ranging compound sentiment scores, we created a master Pandas dataframe with all of our stored information in order to succinctly visualize trends found between clusters.

> *Master Dataframe:* We created 'combined_df' to hold all of the processed information and results of our ML implementation. This dataframe spanned 46 total columns with a plethora of numerical data points.
> *Visualization:* We plotted numerous histograms by comparing compound sentiment score to nearly all of the columns present in *combined_df*. We used these visualizations as the basis for our overarching business strategic decisions.

What was found?

Note: We will be noting clusters by numbers, assuming the reader has analyzed & understood related cluster scores.

Finding #1: Cluster 1 (worst score) is composed primarily of chats with non-TEMO referrers meaning that TEMO Sunrooms' targeted campaigns (not Paid Search or other Google campaigns) successfully fulfill their goal of bringing willing customers to the company's website. **Figure 6.**

Finding #2: Clusters 2, 3, and 5 (top 3 scores) had chats occurring most frequently on Monday through Thursday and Saturday through Sunday. Furthermore, Cluster 1 had the majority of its chats situated around Friday which is statistically the day yielding the worst sentiment scores. This information allows TEMO Sunrooms to launch an internal investigation into possible targeting campaigns aimed around Fridays, representatives regularly monitoring chats on Friday, and their visitor base that may be burned out from a week of work. **Figure 7.**

Finding #3: Clusters 0, 2-5 (all above 90% score) all share high chat session frequency between the months of February through April. This may hint at a possible marketing campaign run around the month of March that was incredibly successful at bringing willing customers into the website. Additionally, comparing where marketing funds were allocated during the months of February through April compared to the rest of the year may illustrate a possible miscall in fund redistribution. **Figure 8.**

Finding #4: Cluster 2 (top score) had chats exclusively outside of normal working hours (08:00 - 18:00 M-F) which may hint at the main customer-base TEMO Sunrooms should work to appeal to, working adults tied to employment-related activities throughout the week. This may further hint at the benefits of a targeted campaign detailing TEMO Sunrooms' ability to remove stress and hassle from consumers by letting their free time be spent pursuing what they enjoy. Purposefully targeting this demographic may boost revenue growth. **Figure 9.**

Finding #5: Clusters 2, 3, and 5 (top 3 scores) had chats most frequently labeled as 'Sales' or 'Service' lead types. This shows that the representatives monitoring the chats within these clusters were able to diagnose the purpose of the session and appropriately direct them to the next step. Further, Sales and Service are the two revenue-focused lead types showing that the demographic within these clusters are the ones that tend to drive greater economic growth for TEMO Sunrooms. **Figure 10.**

Finding #6: Clusters 2, 3, and 5 (top 3 scores) had chats where the visitor opted to provide the most information (name, email, zip, and phone) about themselves. In the graph, the total_info represents the number of pieces of information inputted by the visitor. There is a clear distinction between 2, 3, and 5 with the other 3 clusters. Representatives were able to encourage information from these users thereby leading to greater chat satisfaction. Further looking into these demographics may provide an alleyway for TEMO Sunrooms to construct an ideal customer-base. **Figure 11.**

Figures 6-11 displayed in the following pages.

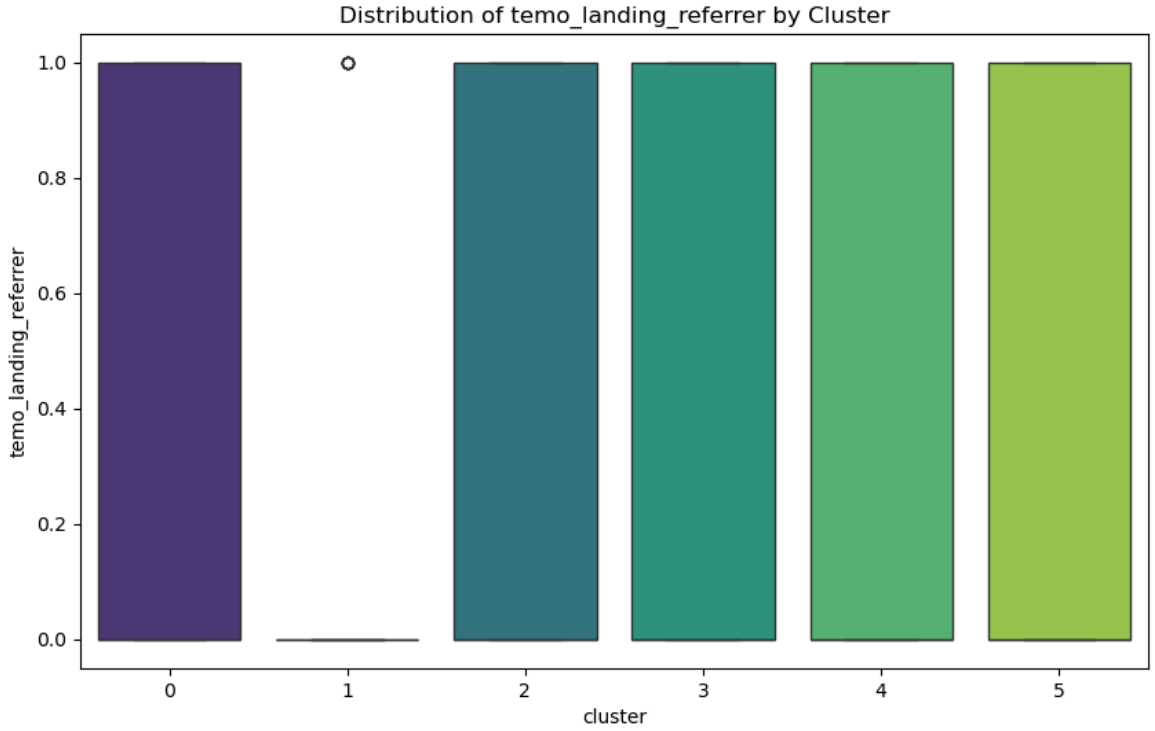

Figure 6: Visitor Sentiment by Landing Referrer Type (1 - TEMO; 0 - OTHER)

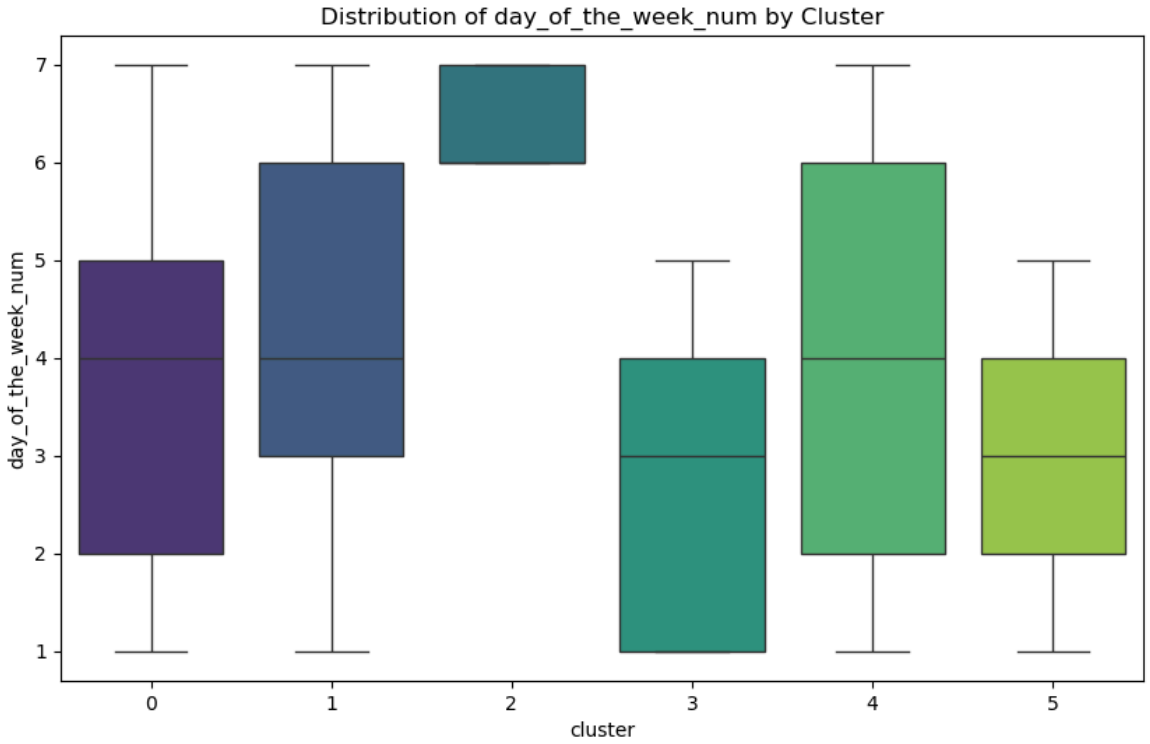

Figure 7: Visitor Sentiment by day of the week (1 - Monday, 2 - Tuesday, …, 7 - Sunday)

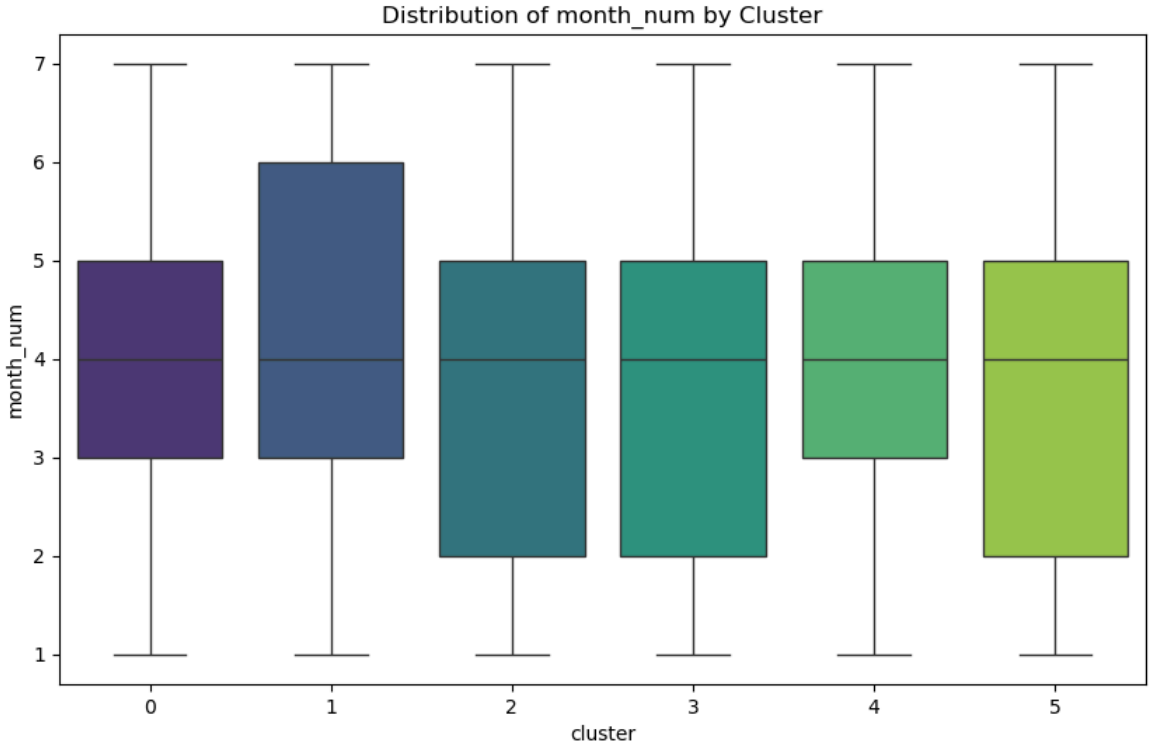

Figure 8: Visitor Sentiment by month (1 - January, 2 - February, …, 7 - July)

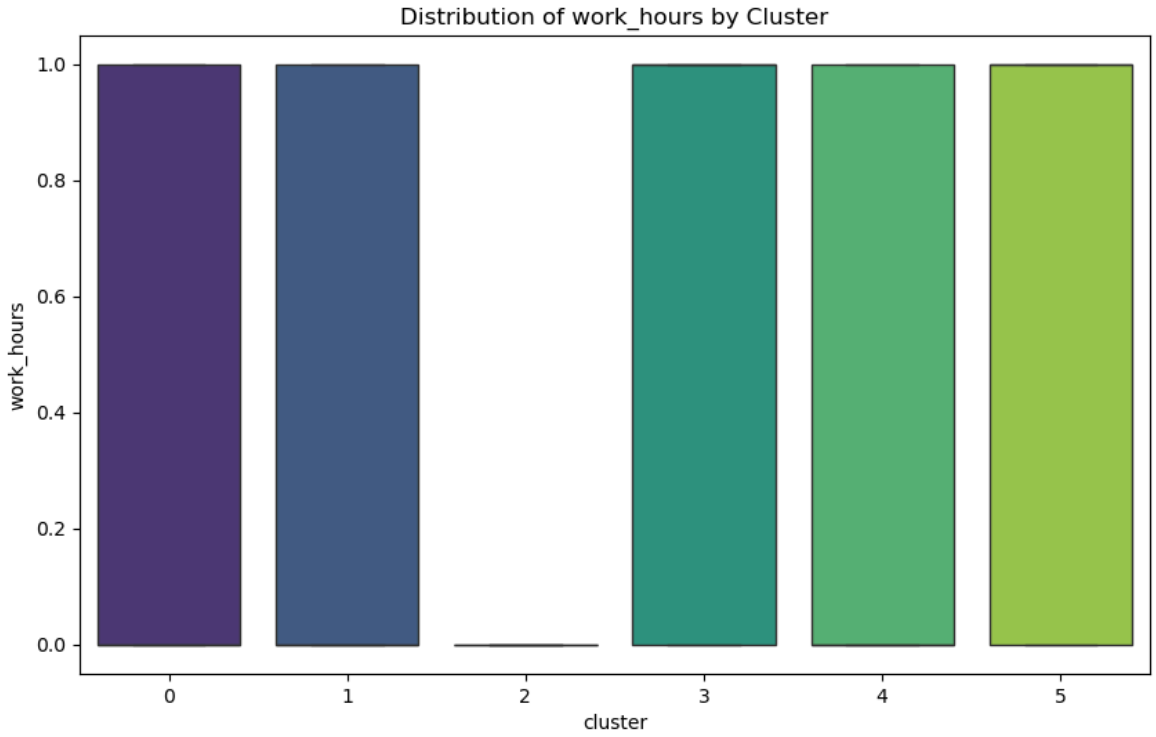

Figure 9: Visitor Sentiment by time of char (1 - inside normal working hours; 0 - outside normal working hours)

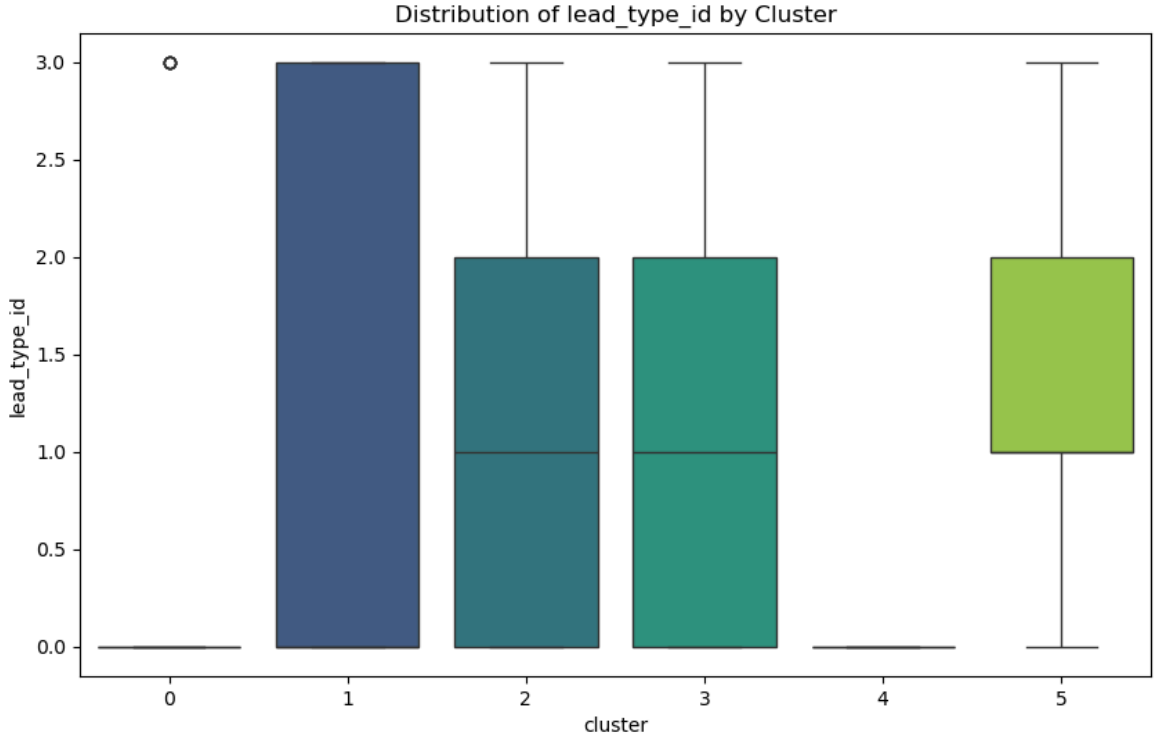

Figure 10: Visitor Sentiment by Lead Type (0 - None; 1 - Sales; 2 - Service; 3 - Other)

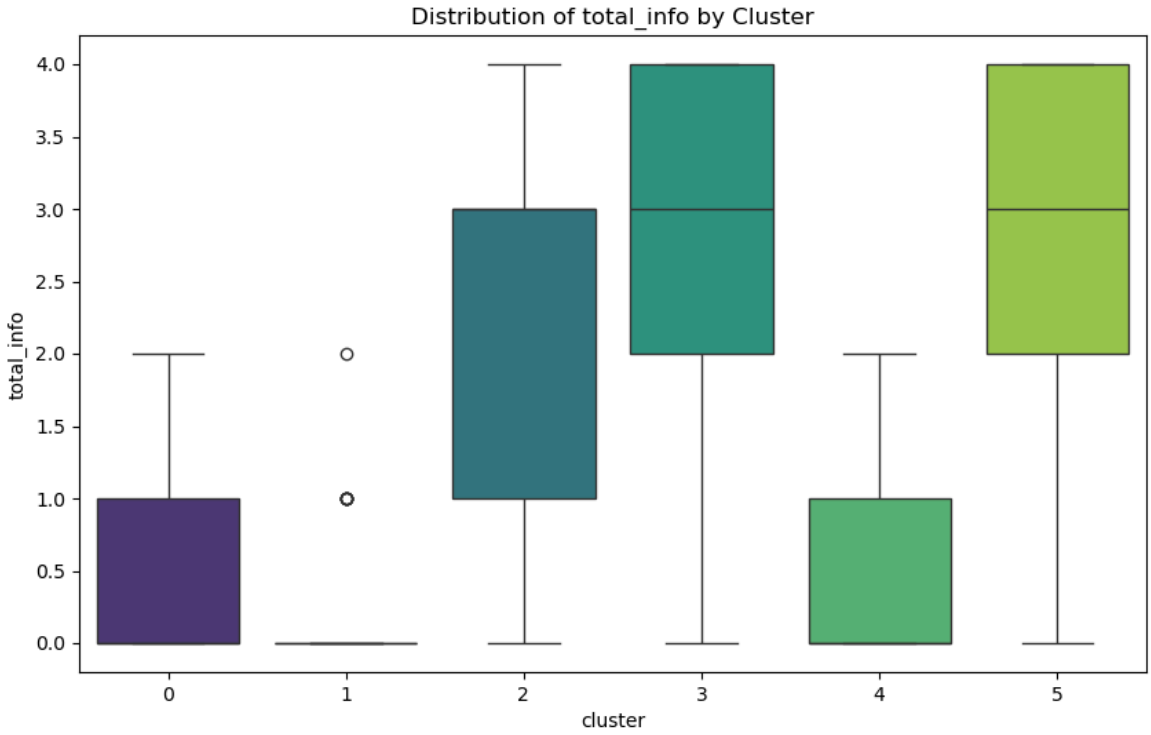

Figure 11: Visitor Sentiment by user-inputted information (tally of name, email, phone, and zip code)