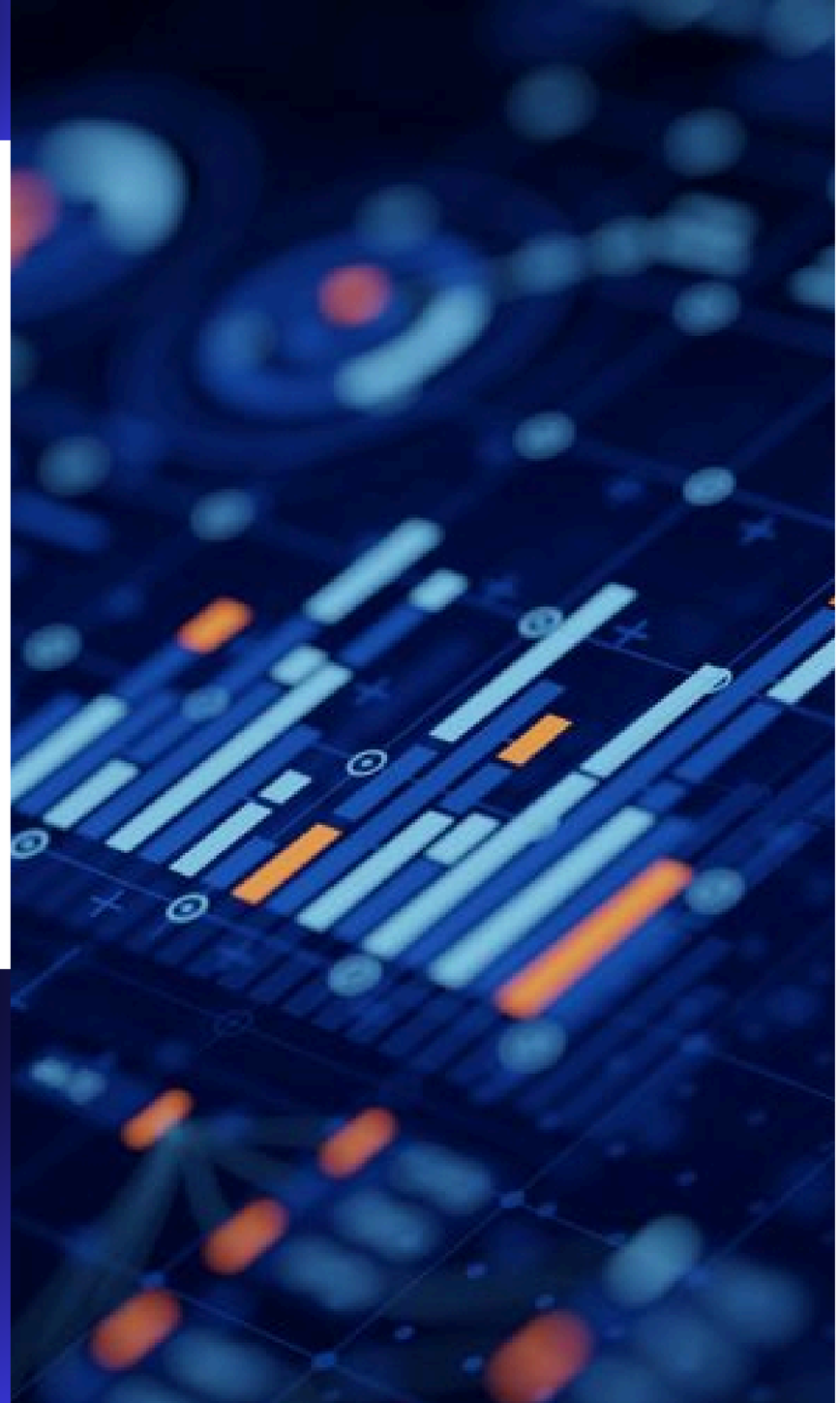
INTERPRETABLE ML FOR CUSTOMER SEGMENTATION

Under the guidance of
Prof. Pabitra Mitra
Prof. Sujoy Bhattacharya

Group Members:
Vishnu Vardhan (21CE3AI13)
Gowtham Chowdary (21CE10084)
Harvin Raj (21ME10090)
Shiva Prasad (21ME31072)



CONTENT

OVERVIEW

This study developed an interpretable machine learning (**IML**)-based approach for customer segmentation based on the importance of product features from online product reviews

Past Studies

Previous studies primarily focused on customer segmentation using demographic, psychographic, and purchase behavior information from transaction databases. They also attempted segmentation based on customer sentiments extracted from online reviews to identify common customer needs

Drawbacks

These approaches struggled to identify customer groups with unsatisfied needs and were limited to sales promotions in marketing. Additionally, sentiment-based segmentation faced challenges due to the diverse and heterogeneous preferences of customers

Our Project

This project proposes an interpretable machine learning approach for customer segmentation, using the importance of product features from online reviews to identify nonlinear relationships and common needs for new product development

DATA COLLECTION & PREPROCESSING

- We collected **11,576** hotel reviews from **four** luxury hotels in London using **Selenium** and **Chrome WebDriver** to automate the clicking and scrolling through pages
- The data was scraped from **Booking.com**, ensuring clean and structured data
- The dataset contains **12 features**, including:

1. Hotel Name
2. Reviewer Name
3. Home Town
4. Traveler Type
5. Room Type
6. Nights Stayed
7. Stayed Month
8. Reviewed Date
9. Review Title
10. Positive Feedback
11. Negative Feedback
12. Review Score

01

Uniformity & Tokenization:

- Converted all text **to lowercase** to ensure consistency and tokenized it into individual words for easier and more precise analysis.
- This turns "The room was clean, and the staff was very helpful! 😊" into "the room was clean, and the staff was very helpful! 😊"

02

Cleaning & Filtering:

- Removed **punctuation** and **stop words** (common words like "and" or "the") that don't contribute significant meaning, refining the text to focus on relevant content
- Now it becomes "room clean staff helpful 😊"

03

Focus & Conversion:

- Applied **lemmatization** to extract key aspects by focusing on nouns, and converted **emojis** to text descriptions to maintain accessibility and context
- This gives us "room clean staff help smiling face"

EXPLORATORY DATA ANALYSIS

Dataset:

Shape of the dataset: (11576, 12)

	Hotel Name	Reviewer Name	Hometown	Traveller Type	Room Type	Nights Stayed	Stayed Month	Reviewed Date
0	The Londoner	Chuvit	Thailand	Solo traveller	King Room	1 night	January 2024	3 January 2024
1	The Londoner	Kay	United Kingdom	Couple	King Room	2 nights	May 2024	15 June 2024
2	The Londoner	Eileen	United Kingdom	Couple	Deluxe King Room	3 nights	June 2024	14 June 2024
3	The Londoner	Andrai	United Kingdom	Couple	King Room	1 night	May 2024	11 June 2024
4	The Londoner	Carman	New Zealand	Solo traveller	Deluxe King Room	5 nights	June 2024	9 June 2024
5	The Londoner	Mradeveci	United Kingdom	Couple	King Room	1 night	May 2024	8 June 2024
6	The Londoner	Rey	Israel	Couple	Deluxe King Room	3 nights	June 2024	6 June 2024
7	The Londoner	Janole	Germany	Couple	Deluxe King City View Room	1 night	April 2024	6 June 2024
8	The Londoner	Elisheva	United Kingdom	Couple	King Room	1 night	May 2024	31 May 2024
9	The Londoner	Kenneth	United Kingdom	Couple	King Room	1 night	May 2024	30 May 2024
10	The Londoner	John	United Kingdom	Couple	King Room	2 nights	May 2024	28 May 2024

Review Title	Positive Feedback	Negative Feedback	Review Score
Front staff is very helpful and friendly. Even in bus	Even surrounding with tourist spot but inside hotel is perfect clean	Breakfast need more varieties and 2 broken running track same time.	10
Exceptional	Absolutely marvellous! Great choice too!	Nothing	10
Very comfortable and very friendly staff.	Location is perfect for theatre land and food and drink.	Nothing	10
Beautiful and luxurious	Everything was amazing!	Nothing!	10
I've found the place to stay on my London trips!!	Staff were great and the hotel facilities excellent.	Tv guide was difficult to see what was on when and where	10
Great location and experience, would stay again.	Location, staff and the wow factor.	The 'residents lounge' seemed a little under staffed and the wait times were quite ap	10
Superb	Beautiful hotel, live music at the lobby, great breakfast, nice pool.	Nowhere in the room to hang towels, not enough space to open a suitcase or unpack	9
Great location with a cozy room.	Very good location. Very modern hotel with many features.	Nothing. It was all perfect.	9
Exceptional	Everything was lovely, rooms, staff, pool	The toilet was unfortunately quite dirty from the last person and very off putting	10
Enjoyable and will return	The location was excellent and the restaurant in the evening was ve	I suppose the price of drinks at the bar but it's the very centre of London so to be exp	8
First time at Londoner and best hotel we've stayec	Everything! Location, facilities and staff.	Nothing.	10

DATA TYPES

	0
Hotel Name	object
Reviewer Name	object
Hometown	object
Traveller Type	object
Room Type	object
Nights Stayed	object
Stayed Month	object
Reviewed Date	object
Review Title	object
Positive Feedbac	object
Negative Feedbac	object
Review Score	float64

UNIQUE VALUES

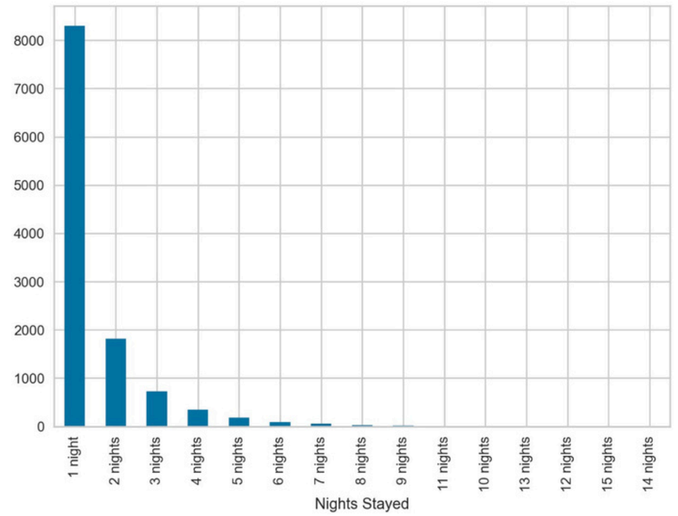
	0
Hotel Name	4
Reviewer Name	3,372
Hometown	110
Traveller Type	5
Room Type	55
Nights Stayed	15
Stayed Month	38
Reviewed Date	1,751
Review Title	8,126
Positive Feedbac	10,351
Negative Feedbac	9,649
Review Score	15

EXPLORATORY DATA ANALYSIS

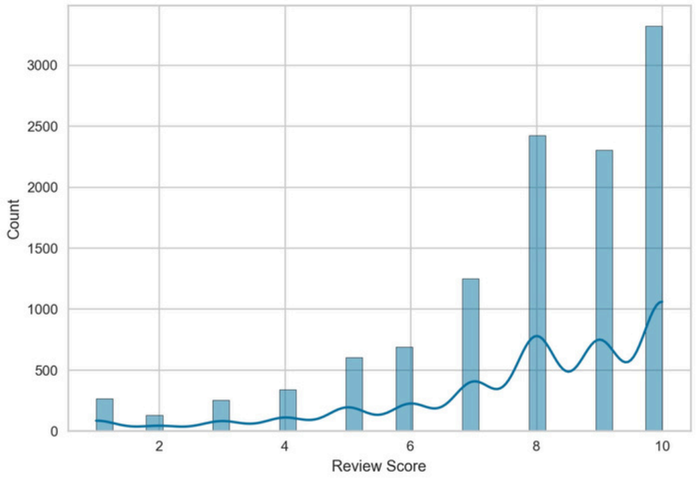
STATISTICS

	Review Score
count	11,576
mean	7.9344
std	2.1688
min	1
25%	7
50%	8
75%	10
max	10

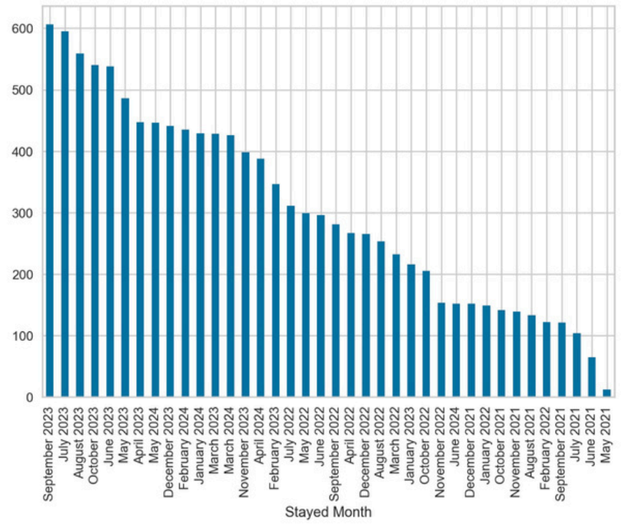
Distribution for Nights Stayed:



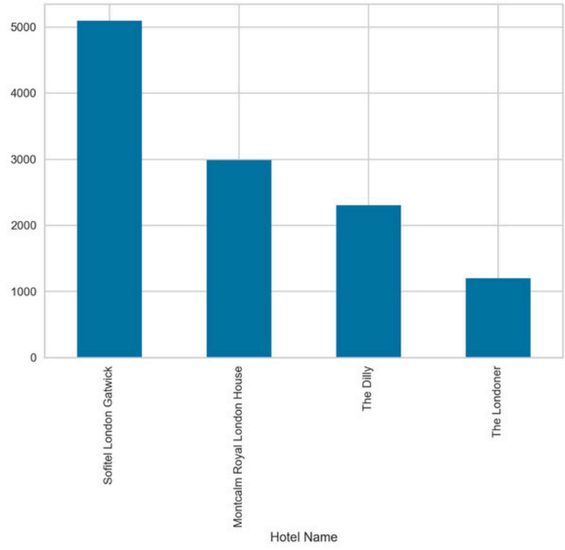
Distribution for Review Score:



Distribution for Stayed Month:



Distribution for Hotel Name:



FEATURE IDENTIFICATION

Chunking & AP Clustering

- **Objective:** Break down the large set of hotel review word vectors into smaller, manageable chunks
- **Process:** For each chunk, the **Affinity Propagation (AP)** clustering algorithm groups similar words based on their **cosine similarity**. The most representative word of each cluster **(exemplar)** is identified

Combining Exemplars

- **Objective:** Gather and unify the exemplars (representative words) from all the chunks to form a comprehensive set
- **Process:** The exemplars and their associated words are collected from the **41 divided chunks**, resulting in a combined dataset ready for further analysis

Hierarchical Clustering

- **Objective:** Further refine the clusters by applying **hierarchical clustering** to the combined exemplars
- **Process:** Calculate the distances between exemplars using Euclidean distance and cluster them using the **Ward method**. This process identifies clusters of similar exemplars across all chunks

Feature Identification

- **Objective:** Identify key features from hotel reviews
- **Process:** For each hierarchical cluster, determine the centroid and find the word closest to it (exemplar). This exemplar represents the key feature, which is then manually reviewed and categorized to extract meaningful features like **"window," "bathroom," and "service"**

AP CLUSTERING

Affinity Propagation is a method used to find groups or clusters in data without needing to specify how many clusters there should be. It works by identifying "exemplars," which are representative points that best describe each group

How It Works:

- **Similarity:** The algorithm starts by calculating how similar each data point is to every other point in the dataset

- **Responsibility:** Each data point considers how well-suited another point would be as its exemplar, assigning a "responsibility" score that reflects this suitability

- **Availability:** Simultaneously, each point also evaluates how appropriate it is to be an exemplar for other points, which is expressed as an "availability" score

- **Exemplars:** By combining responsibility and availability scores, the algorithm identifies the most representative points (exemplars) for each group

- **Grouping:** Finally, it groups other similar data points around these exemplars, forming clusters

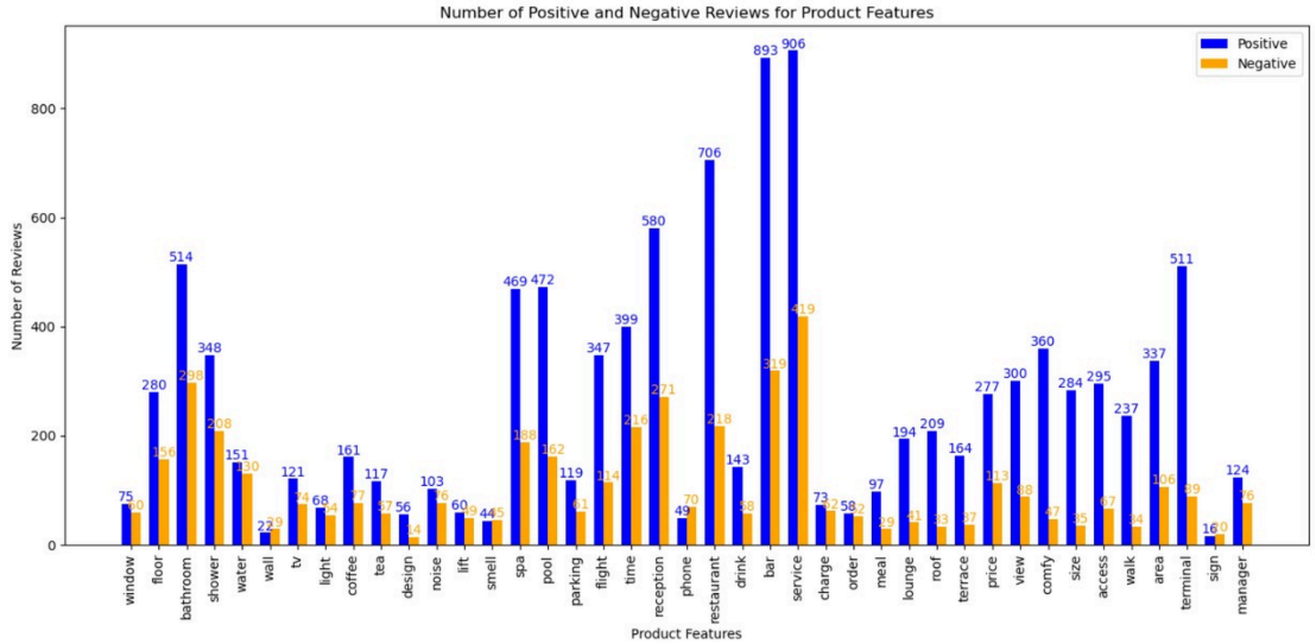
Key Points:

- **No Need for Predefined Clusters:** You don’t need to guess the number of clusters ahead of time

- **Automatic Finding of Clusters:** The algorithm decides the number of clusters based on the data

SENTIMENT ESTIMATION

- Using **VADER**, the sentiment (positive, negative, or neutral) of sentences mentioning these features is analyzed, with sentiment scores ranging from **-1 (most negative)** to **+1 (most positive)**
- The sentiment scores for each feature are combined to determine the overall feedback for that feature, which is then visualized in a bar graph showing the number of positive and negative reviews
- Reviews are labeled as "positive" or "negative" based on both the review score and the sentiment scores, providing a more accurate understanding of customer feedback



No. of positive and negative reviews in each feature

Hotel Review: "I had a wonderful stay at the Grandview Hotel. The staff were incredibly friendly and the room was spotless. However, the noise from the street was quite disturbing at night."

Positive Sentiment: "I had a wonderful stay at the Grandview Hotel. The staff were incredibly friendly and the room was spotless."

Negative Sentiment: "the noise from the street was quite disturbing at night."

Overall Sentiment Score: Mixed (positive with a minor negative aspect)

ML CLASSIFIERS

- Various machine learning algorithms—**DesicionTrees, RandomForest, LGBM, XGBoost, and CatBoost**—are considered to obtain the optimal classifiers. The classifiers are developed to predict **star ratings** based on the **sentiment scores of product features.**
- In the construction of the machine learning models the **input** variables are the **sentiment scores** of the product features (i.e., pf_1, pf_2, …, pf_i) in the 1, 2, …, M review). The **output variables** are the labels of the positive and negative **star ratings** corresponding to the reviews
- **Nested cross-validation** is performed to optimize the hyperparameters of various classifiers and identify the best-performing model from the entire dataset.
- The prediction performance of each classifier is evaluated using the **F1-score.** The top-performing classifier among the five is then selected for analysis using the **SHAP method.**

Decision Tree

(F1 Score: 0.9286)

Random Forest

(F1 Score: 0.9298)

LGBM

(F1 Score: 0.9199)

XgBoost

(F1 Score: 0.9295)

CatBoost

(F1 Score: 0.9291)

K×K′ CROSS VALIDATION

- The optimization is aimed to over come the bias in performance evaluation depending on the configuration of the **training, validation, and test sets**.
- The nested cross-validation constitutes inner and outer loops of the entire dataset, and a **K-fold** cross-validation is conducted in each loop
- In the outer loop, the entire dataset is first randomly partitioned into **K equal-sized training and test sets**.
- Subsequently, each training set of the outer loop in the inner loop is randomly divided into **K′** equal-sized training and validation sets.

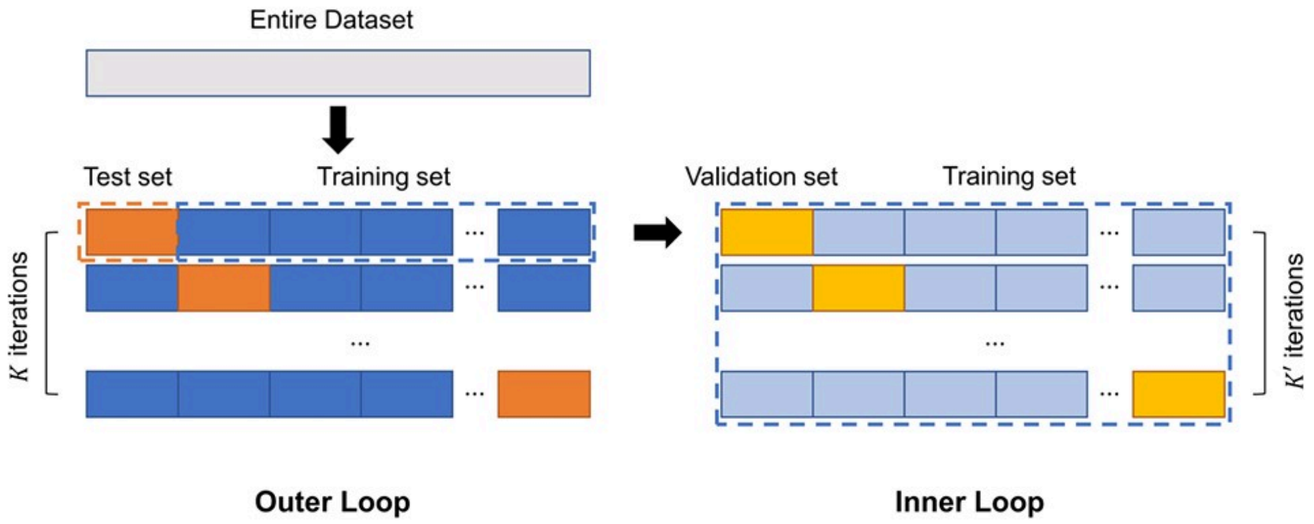


Fig. 3. Nested cross-validation.

- **K′-1** sub-samples are used as the training sets to train a classifier, and the remaining single sub-sample is used as the **validation set** to optimize the **hyperparameters** of the classifier.
- The performance of the classifier is evaluated using the **test set** corresponding to the training set of the outer loop. The prediction performance of each classifier is calculated using the **f1-score**
- Consequently, **K×K′** optimal machine learning classifiers with the best f1-score are obtained from the **K×K′** training sets.

PREDICTION PERFORMANCE - F1 SCORES

Outer Fold	Inner Fold	DecisionTree	RandomForest	CatBoost	XGBoost	LightGBM
1	1'	0.933642	0.936023	0.934536	0.93505	0.863416
	2'	0.932357	0.934802	0.934426	0.935248	0.980997
	3'	0.927426	0.928405	0.929032	0.929727	0.981848
	4'	0.918649	0.920235	0.918563	0.918984	0.807321
	5'	0.928405	0.929733	0.929245	0.929619	0.89604
	Avg	0.934587	0.934838	0.933749	0.935611	0.95953

Outer Fold	Inner Fold	DecisionTree	RandomForest	CatBoost	XGBoost	LightGBM
2	1'	0.931045	0.93314	0.930653	0.931571	0.967562
	2'	0.932252	0.932909	0.933761	0.932825	0.985475
	3'	0.930246	0.931094	0.932512	0.932864	0.987695
	4'	0.929774	0.930314	0.932313	0.931289	0.801831
	5'	0.918452	0.922537	0.923764	0.923214	0.897823
	Avg	0.928511	0.931304	0.931031	0.930601	0.886918

PREDICTION PERFORMANCE - F1 SCORES

Outer Fold	Inner Fold	DecisionTree	RandomForest	CatBoost	XGBoost	LightGBM
3	**1'**	0.934574	0.935344	0.933726	0.935088	0.957041
	2'	0.930192	0.931983	0.933921	0.930862	0.909368
	3'	0.927114	0.927172	0.928698	0.928019	0.943396
	4'	0.928968	0.928488	0.929835	0.929519	0.81
	5'	0.926986	0.928112	0.92836	0.928445	0.895596
	Avg	0.930784	0.931403	0.930661	0.930188	0.985088

Outer Fold	Inner Fold	DecisionTree	RandomForest	CatBoost	XGBoost	LightGBM
4	**1'**	0.933178	0.934877	0.936805	0.93719	0.961961
	2'	0.933333	0.93437	0.934072	0.93447	0.904704
	3'	0.932636	0.934455	0.936369	0.93719	0.954909
	4'	0.928154	0.929299	0.933608	0.930396	0.989905
	5'	0.926843	0.92934	0.927059	0.92657	0.899516
	Avg	0.921355	0.922796	0.921986	0.92112	0.856135

PREDICTION PERFORMANCE - F1 SCORES

Outer Fold	Inner Fold	DecisionTree	RandomForest	CatBoost	XGBoost	LightGBM
5	**1'**	0.936058	0.937428	0.937738	0.937592	0.955309
	2'	0.934574	0.935409	0.930884	0.932039	0.903206
	3'	0.929504	0.930151	0.931879	0.932544	0.946459
	4'	0.923617	0.927114	0.926958	0.926887	0.985197
	5'	0.926887	0.925547	0.925576	0.925576	0.811966
	Avg	0.928013	0.928994	0.928473	0.930225	0.912205

Metric	DecisionTree	RandomForest	CatBoost	XGBoost	LightGBM
Final Avg	0.92865	**0.929867**	0.92918	0.929549	0.919975

- Since **Random Forest** has the **highest F1 score**, it is the most accurate model in your set. Using SHAP on this model will yield explanations based on the most reliable predictions
- Random Forest is an **ensemble method** that combines multiple decision trees, leading to complex decision boundaries. SHAP is particularly useful in breaking down these complex models by attributing the contribution of each feature to the model's predictions

FEATURE IMPORTANCE

What is SHAP?

SHAP **(SHapley Additive exPlanations)** is a tool from **game theory** that helps us understand how each feature of our data impacts the predictions made by our machine learning model. In our project, SHAP shows how different aspects of hotel reviews, like room cleanliness and staff friendliness, affect the star ratings.

Why We Use SHAP?

We use SHAP because it provides a clear and fair way to see how each feature influences the model's predictions. It considers all possible interactions between features, making it especially useful for complex models.

How SHAP Works ?

01

Base Value Calculation

SHAP starts with the average rating across all reviews. For example, if the average rating is 3.5 stars, this is the base value from which feature contributions are measured

02

Feature Impact Measurement

SHAP calculates how each feature (like "clean room" or "friendly staff") influences the final prediction by comparing it to the base value. It shows how each feature shifts the prediction away from the average rating

03

Contribution Distribution

Instead of simply adding or subtracting stars, SHAP provides a detailed breakdown of how much each feature contributes to or detracts from the prediction relative to the base value, giving a clear view of its impact

Formula for SHAP Value::

$$\phi_{pfi}(v) = \sum_{S \subseteq F : i \notin S} \frac{|S|!(|F| - |S| - 1)!}{|F|!} \left(v(S \cup \{pfi\}) - v(S)\right)$$

where:

- $\phi_{pfi}(v)$ is the SHAP value for feature pfi,
- S represents a subset of features not including pfi,
- F is the set of all features,
- $v(S \cup \{pfi\})$ is the model's prediction with pfi included,
- $v(S)$ is the model's prediction without pfi.

Formula for Feature Importance in Each Review:

For each review im, the importance value of a feature based on SHAP is:

$$\text{Imp}_{im}^{k'} = |\text{SHAP}_{im}^{k'}|$$

Where:

- $\text{Imp}_{im}^{k'}$ is the importance value of a feature in review im from classifier k'.
- $|\text{SHAP}_{im}^{k'}|$ is the absolute SHAP value for the feature in review im from classifier k'.

Overall Feature Importance for Each Review:

The overall importance of features in review im is calculated by averaging the importance values from all K' classifiers, weighted by their performance:

$$\text{Imp}_{im} = \sum_{k'=1}^{K'} w_{k'} \text{Imp}_{im}^{k'}$$

Where:

- Imp_{im} is the aggregated importance value for review im.
- $w_{k'}$ is the weight for classifier k', calculated as:

$$w_{k'} = \frac{w_{k'}}{\sum_{k'=1}^{K'} w_{k'}}$$

The Process We Followed...

- **Model Training with Nested Cross-Validation:** We trained a Random Forest model using nested cross-validation, splitting our data into multiple parts to find the best model settings. The goal was to achieve the highest accuracy, measured by the F1 score

- **Applying SHAP:** After finding the best model, we used SHAP to see how the model makes predictions. SHAP values showed how much each feature in a review, like room cleanliness or staff friendliness, contributed to the predicted star rating

- **Understanding SHAP with an Example:** For a review mentioning a clean room, friendly staff, and a convenient location, if the model predicted a 4-star rating:
  - SHAP starts with the average rating, say 3.5 stars
  - Clean Room might add 0.3 stars, Friendly Staff 0.2 stars, and Convenient Location 0.1 stars, leading to a 4.1-star prediction

- **Analysis of Results:** By averaging SHAP values across all reviews, we identified which features most influenced star ratings, helping us understand customer preferences and areas for hotel service improvement.

CUSTOMER SEGMENTATION

What We Did ?

We used **K-Means** clustering to group hotel reviews based on the importance of features like room cleanliness and staff friendliness.

Why We Did It ?

- **Importance:** We measured how crucial each feature is to customer satisfaction. For example, if "clean room" is mentioned frequently, it gets a higher importance score.
- **Satisfaction:** We checked how happy customers are with each feature. If "friendly staff" is important but customers aren't satisfied, it shows in the satisfaction scores.
- **Opportunity:** We combined importance and satisfaction scores to find areas where improvements could significantly boost customer satisfaction

FORMULA USED :

$$\text{Opportunity}_c = \text{Importance}_i + \max(\text{Importance}_i - \text{Satisfaction}_i, 0)$$

Explanation:

- **Importance**$_i$**:** The average importance of a feature i in a specific customer cluster c.
- **Satisfaction**$_i$**:** The average satisfaction with that feature i in the same cluster.
- **Opportunity**$_c$**:** This score highlights features that are important but not meeting customer expectations, indicating where improvements are needed.

METHODS :

- **Scaling Data:** We standardized the data to ensure accuracy.
- **Clustering:** We grouped reviews into clusters based on feature importance scores
- **Calculating Scores:** We averaged importance, satisfaction, and opportunity scores for each feature within each cluster

Example:

Cluster 1: High importance on "Clean Room" but low satisfaction
Cluster 2: High importance and satisfaction with "Friendly Staff"
Cluster 3: Moderate importance and satisfaction with "Convenient Location"

This analysis shows where improvements can have the most impact. For example, if **Cluster 1** has a **high opportunity score** for "Clean Room," enhancing this feature could greatly increase satisfaction

TOP COSTUMER FEATURES (1-9)

Feature	Importance	Satisfaction	Oppurtunity	No. of Reviews
manager	7.026059801	0.050251256	14.00186835	199
bar	4.810433046	5.520675602	4.810433046	87
service	4.971487821	0.279072142	9.6639035	81
bathroom	6.288672671	0.161290323	12.41605502	62
terminal	5.640056611	0.428466618	10.8516466	56
tv	5.041787534	0.188679245	9.894895822	53
roof	4.927585476	0.2	9.655170951	50
pool	5.143299102	2.038842634	8.24775557	49
price	5.566792302	0.416400951	10.71718365	47

Above clusters are extracted using K means Clustering

TOP COSTUMER FEATURES (9-18)

Feature	Importance	Satisfaction	Oppurtunity	No. of Reviews
floor	4.26465555	0.222222222	8.307088878	45
shower	3.765113909	0.243902439	7.28632538	41
lounge	4.046137417	0.322580645	7.769694188	31
bathroom	4.880026411	0.344827586	9.415225236	29
spa	4.16403775	0.379003559	7.949071942	28
flight	6.382528881	0.398989449	12.36606831	28
service	6.382307477	1.635230441	11.12938451	26
lounge	5.051049432	0.4	9.702098863	25
lift	3.986440377	9.6	3.986440377	25

Above clusters are extracted using K means Clustering

THANK YOU