Mastering Customer Segmentation with LLMs

Author: Aryan Sharma

Overview

This document provides a comprehensive guide to mastering customer segmentation using advanced clustering techniques, including K-means, K-prototype, and a hybrid approach that integrates Language Models (LLMs) with K-means. Leveraging a publicly available dataset from Kaggle's "Banking Dataset — Marketing Targets," it details the entire process from data preprocessing to model evaluation. The K-means method is enhanced with sophisticated preprocessing steps, outlier detection, and optimal cluster determination using the Elbow Method and Silhouette Analysis. The K-prototype technique extends these capabilities to handle mixed numerical and categorical data. The innovative LLM + K-means hybrid approach combines the strengths of natural language processing and traditional clustering, transforming structured data into descriptive sentences and vectorizing them with the GloVe model for enriched insights. Dimensionality reduction techniques such as PCA, t-SNE, and MCA are employed to visualize high-dimensional data, ensuring robust and meaningful segmentation. The document emphasizes the importance of thorough preprocessing and validation, showcasing how integrating advanced NLP techniques with clustering methods can significantly enhance customer segmentation strategies.

Table of Contents

1. Introduction

This project aims to provide advanced techniques for customer segmentation, enhancing traditional clustering models with sophisticated and innovative approaches. Utilizing a variety of methodologies, including K-means, K-prototype, and a hybrid method that integrates Language Models (LLMs) with K-means, this study offers a comprehensive exploration of cutting-edge techniques in customer segmentation. It leverages a publicly available dataset from Kaggle's "Banking Dataset — Marketing Targets" to demonstrate practical applications and effectiveness. Intended for data scientists and analysts, this project seeks to deepen their understanding and capabilities in addressing complex clustering problems by providing detailed step-by-step instructions on preprocessing, modeling, and evaluating clustering techniques. By incorporating advanced natural language processing techniques, such as text vectorization using the GloVe model, alongside traditional clustering methods, the project aims to produce more accurate and insightful customer segments. This holistic approach not only enhances the robustness and accuracy of clustering models but also paves the way for innovative solutions to complex customer segmentation challenges, making it a valuable resource for professionals seeking to advance their expertise in this domain.

2. Data Description

The dataset used is from Kaggle's "Banking Dataset — Marketing Targets". It includes information about customers with various numerical and categorical fields. The project focuses on the following columns:

- **age**: Numeric
- **job**: Categorical
- **marital**: Categorical
- **education**: Categorical
- **default**: Binary
- **balance**: Numeric
- **housing**: Binary
- **loan**: Binary

The dataset is preprocessed to include only the first eight columns, and outliers are detected and handled using the ECOD method from the PyOD library.

3. Methodologies

Method 1: K-means

K-means is a popular clustering algorithm that requires specifying the number of clusters. Key steps include:

- **Preprocessing**: Convert categorical variables to numerical using OneHotEncoder and OrdinalEncoder. Normalize numerical variables using PowerTransformer.
- **Outlier Detection**: Use the ECOD method to identify and remove outliers.
- **Elbow Method**: Determine the optimal number of clusters by plotting the distortion for different k values.
- **Silhouette Analysis**: Validate the chosen number of clusters using silhouette scores.

Preprocessing:

- **One-hot encoding for binary variables (`default`, `housing`, `loan`)**: Convert these variables into binary numeric format.

- **One-hot encoding for multi-level categorical variables (`job`, `marital`)**: Use `model.matrix` to create dummy variables for each level of the categorical variable, excluding the intercept.
- **Ordinal encoding for `education`**: Convert the `education` variable to an ordered factor and then to numeric based on the defined levels.
- **Scaling numeric variables (`age`, `balance`)**: Standardize these variables so they have a mean of 0 and a standard deviation of 1.
- **Outlier identification**:

  - Loop through specified columns (`age`, `balance`).
  - Calculate the first quartile (Q1), third quartile (Q3), and IQR.
  - Determine the upper and lower limits for identifying outliers.
  - Collect indices of rows with outliers.

- **Removing outliers**: Use the collected indices to remove rows containing outliers from the dataset.

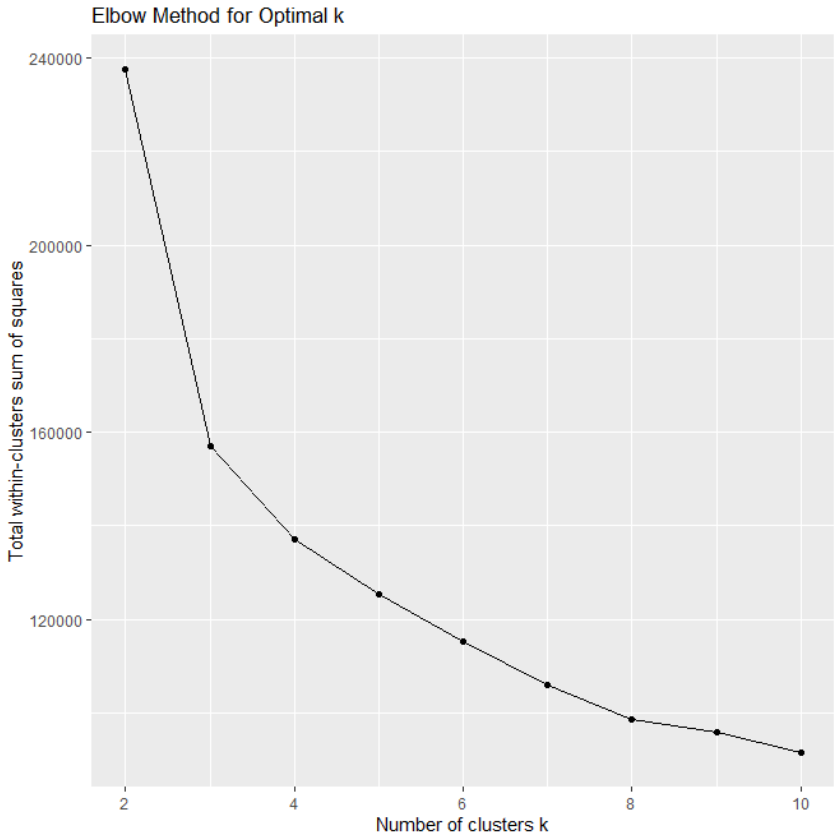
The initial dataset has 45211 observations, and after removing the outliers based on the scaled columns of balance and age we get a dataset with 40108 observations.

Determining the value of K

Elbow Method

To determine the optimal number of clusters for a dataset by using the Elbow Method. We first preprocess the data using the CLARA algorithm to handle large datasets, then computes the total within-cluster sum of squares (WSS) for different values of `k` (number of clusters). The WSS

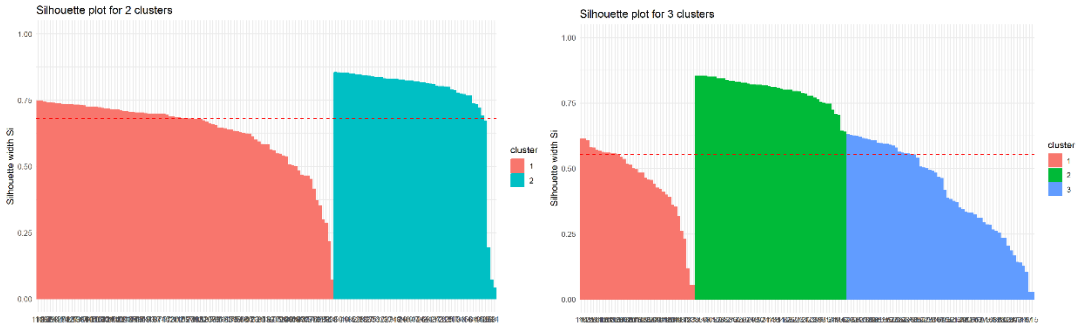
values are plotted to create an Elbow plot, which helps in identifying the optimal `k` value.

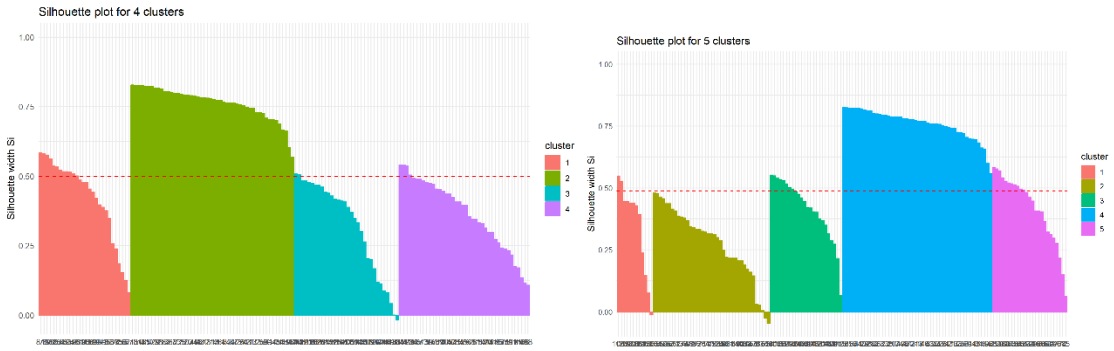


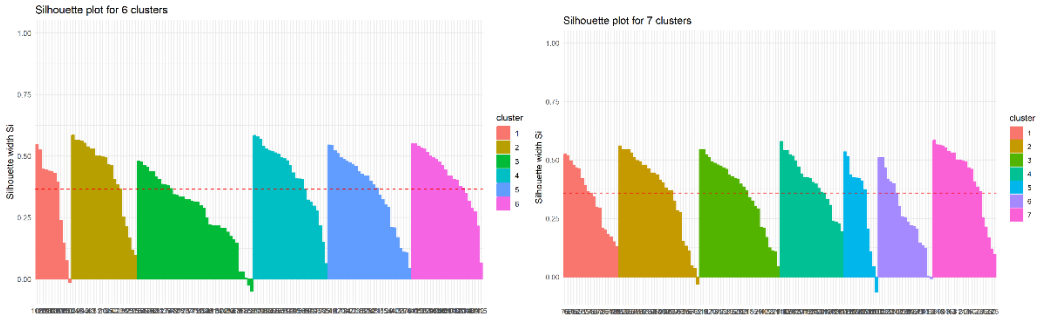
Silhouette plot

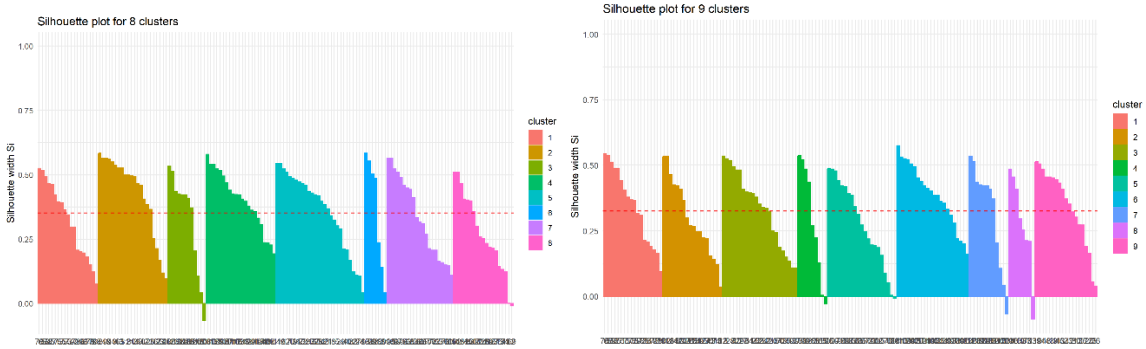
A graphical representation used to interpret and validate the consistency within clusters of data.

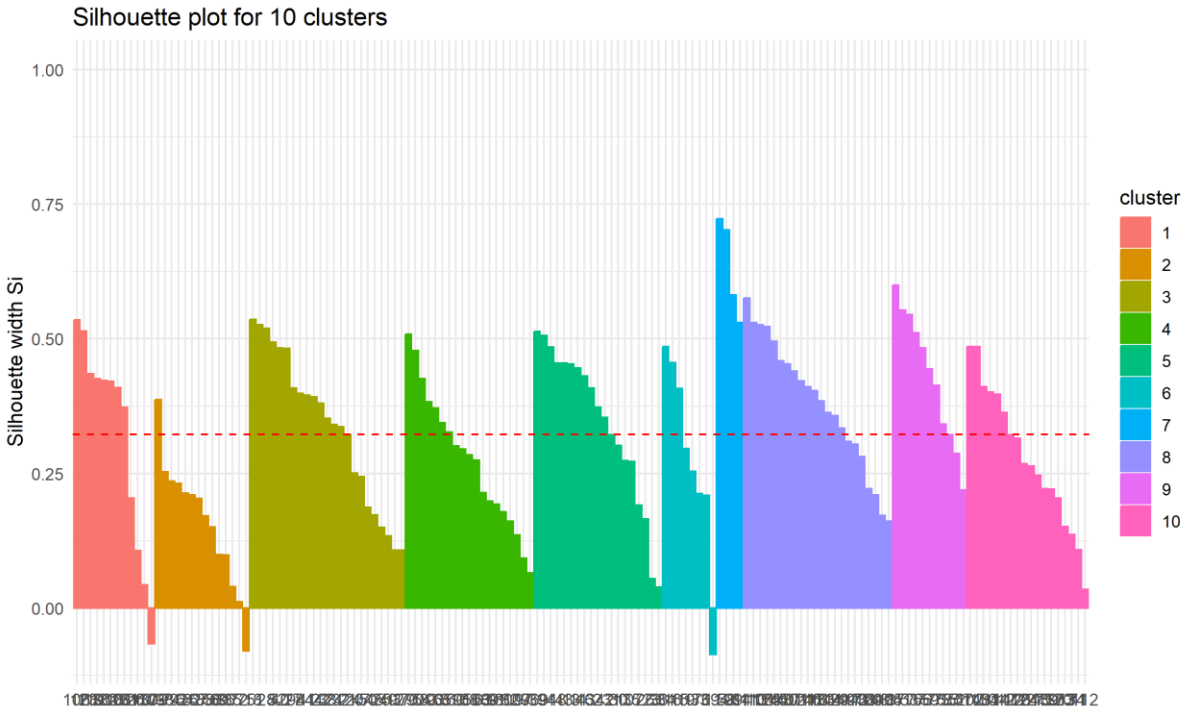
- Each bar represents a data point.
- The length of each bar indicates the silhouette value of that data point.
- A higher silhouette value indicates that the data point is well matched to its own cluster and poorly matched to neighboring clusters.
- The average silhouette width indicates the overall quality of the clustering




Silhouette plot for 4 clusters
Silhouette width Si
cluster
1
2
3
4
Silhouette plot for 5 clusters
Silhouette width Si
cluster
1
2
3
4
5


Silhouette plot for 6 clusters
Silhouette width Si
cluster
1
2
3
4
5
6
Silhouette plot for 7 clusters
Silhouette width Si
cluster
1
2
3
4
5
6
7


Silhouette plot for 8 clusters
Silhouette width Si
cluster
1
2
3
4
5
6
7
8
Silhouette plot for 9 clusters
Silhouette width Si
cluster
1
2
3
4
5
6
7
8
9

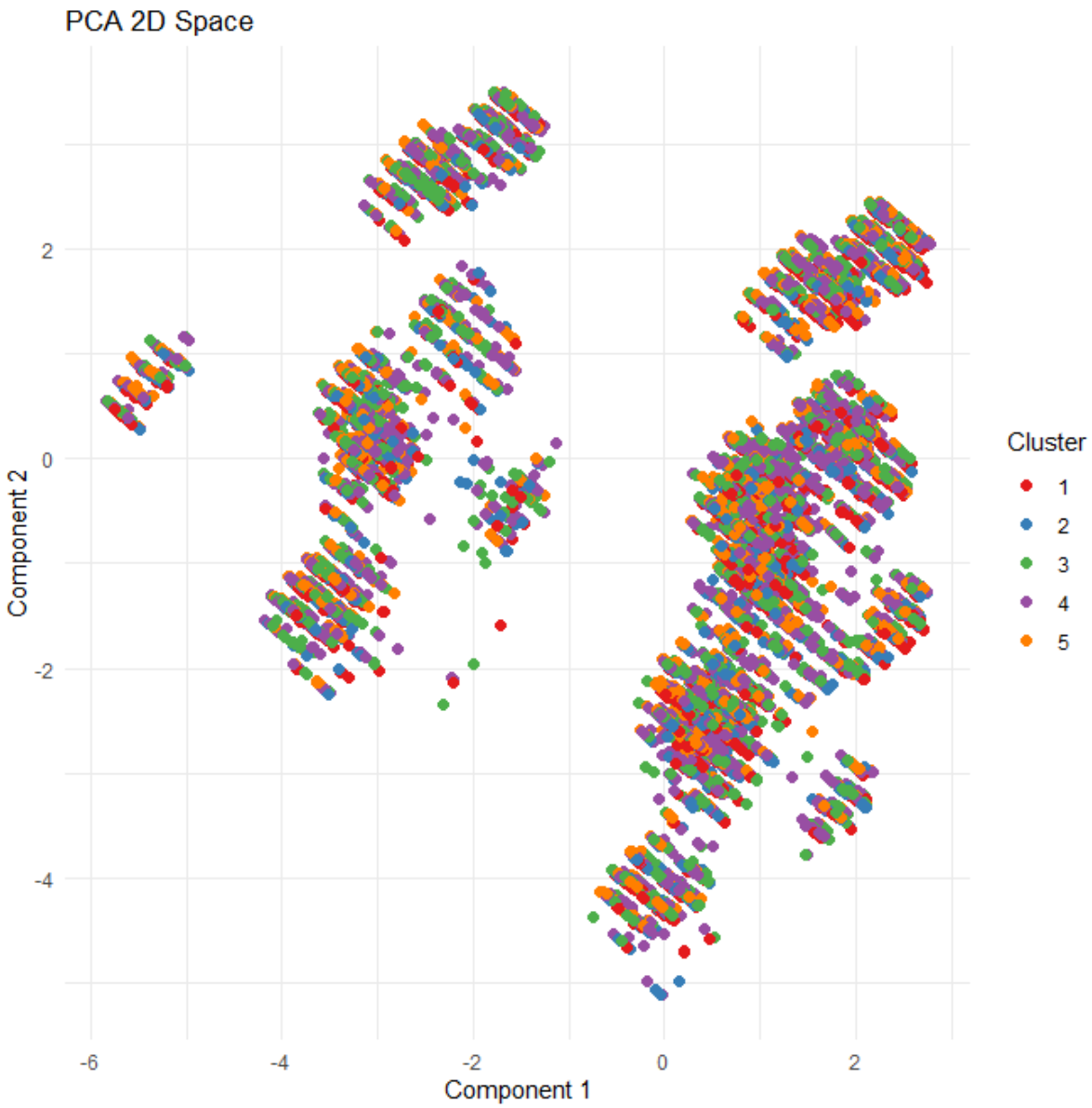


After seeing the plots for k=2 to k = 10. We get 9 graphs and based on the graphs we can say k value 5 & 6 would be the best k value for the dataset.

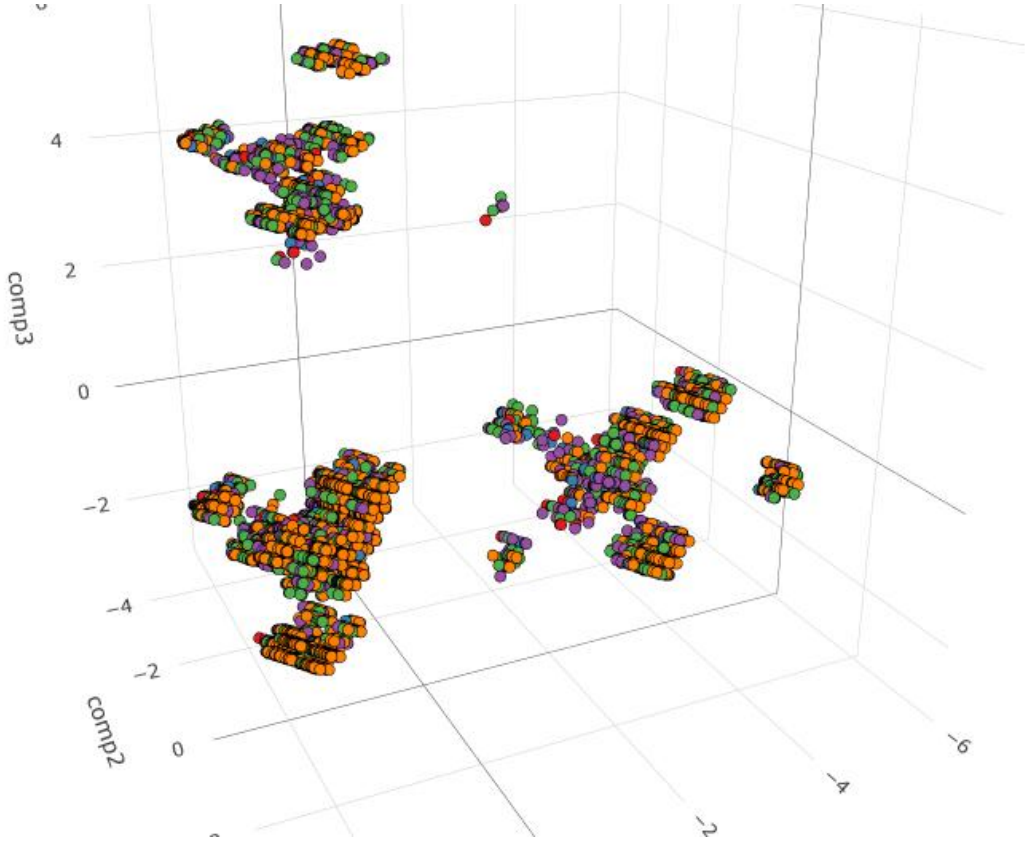
```
> db_index <- intCriteria(as.matrix(data_no_outliers), clusters_predict, "Davies_Bouldin")
> davies_bouldin <- db_index[[1]]  # Extract the Davies Bouldin score
> cat("Davies Bouldin score:", davies_bouldin, "\n")
Davies Bouldin score: 0.5113154
> # Compute Calinski Harabasz score
> ch_index <- intCriteria(as.matrix(data_no_outliers), clusters_predict, "Calinski_Harabasz")
> calinski_score <- ch_index[[1]]  # Extract the Calinski Harabasz score
> cat("Calinski Score:", calinski_score, "\n")
Calinski Score: 206573.1
> # Compute Silhouette score
> sil <- silhouette(clusters_predict, dist(data_no_outliers))
> silhouette_score <- mean(sil[, 3])
> cat("Silhouette Score:", silhouette_score, "\n")
Silhouette Score: 0.5800217
```

Our model performance is suboptimal, as indicated by Davies' score, which reveals minimal separation between clusters.

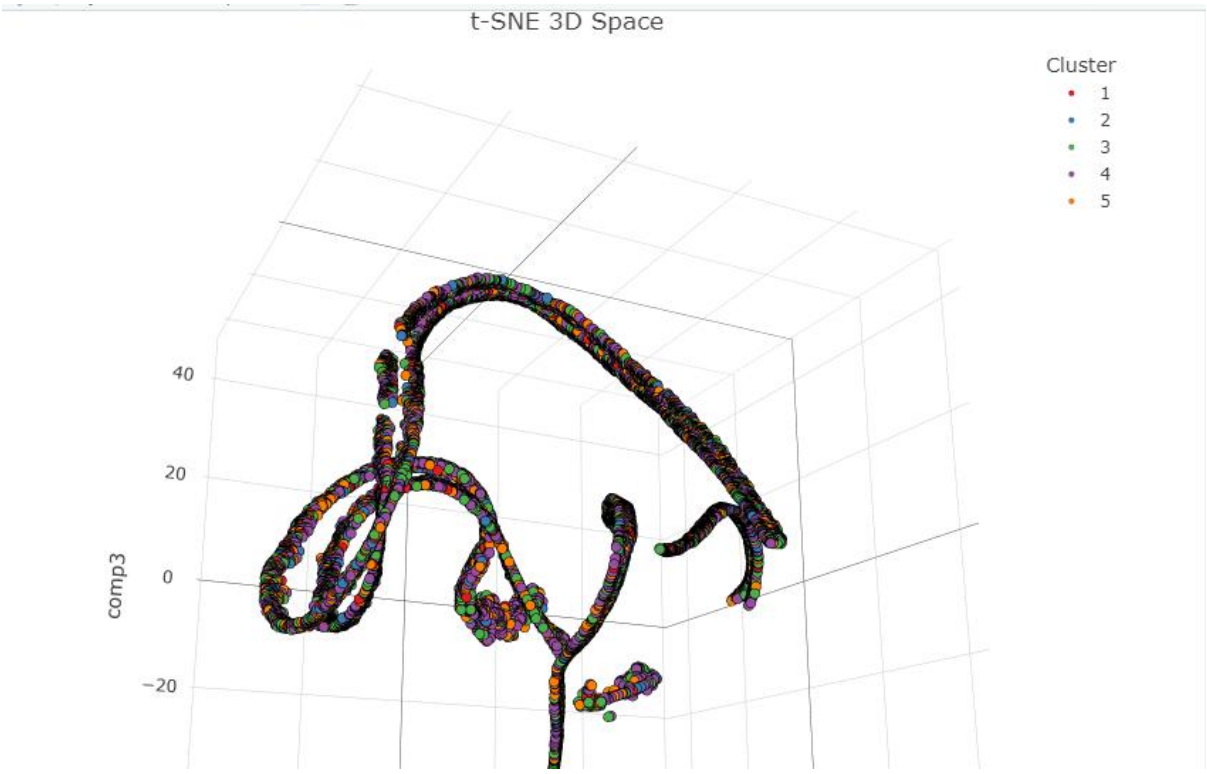
This could be attributed to various factors, but it is crucial to remember that the effectiveness of a model is inherently dependent on the quality of the data. Without adequate predictive power in the data, achieving outstanding results is unrealistic.



2D PCA clustering



3D visualtisation of clustering via PCA



3D visualisation of clustering via Tsne

We will use the LGBMClassifier, which is a robust model capable of handling both categorical and numerical variables effectively. After training this new model, we will utilize the SHAP library to evaluate the importance of each feature in the prediction. The code for this analysis is forthcoming.

Method 2: K-prototype

K-prototype clustering is suitable for mixed numerical and categorical data. It extends K-means by handling categorical data effectively.

Data Encoding and Scaling

Categorical variables (`default`, `housing`, `loan`) are encoded into binary formats using one-hot encoding. Numerical variables (`age`, `balance`) are scaled to have a mean of zero and a standard deviation of one. This normalization is crucial for many statistical models to perform correctly.

Outlier Detection and Removal

Outliers in the `age` and `balance` columns are identified using the Interquartile Range (IQR) method. Observations that fall outside 1.5 times the IQR below the first quartile or above the third quartile are considered outliers and are removed from the training data. This is done to prevent extreme values from distorting the results of the analysis.

Data Frame Cleanup

Duplicates in outlier indices are removed to ensure that each outlier is only removed once. A new data frame without these outliers, `train_data_no_outliers`, is created and summarized to check the data's structure after cleaning.

Factor Conversion

Certain columns are converted to factors, which is necessary for categorical data to be properly handled in statistical models. This includes the `job`, `marital`, and `education` columns.

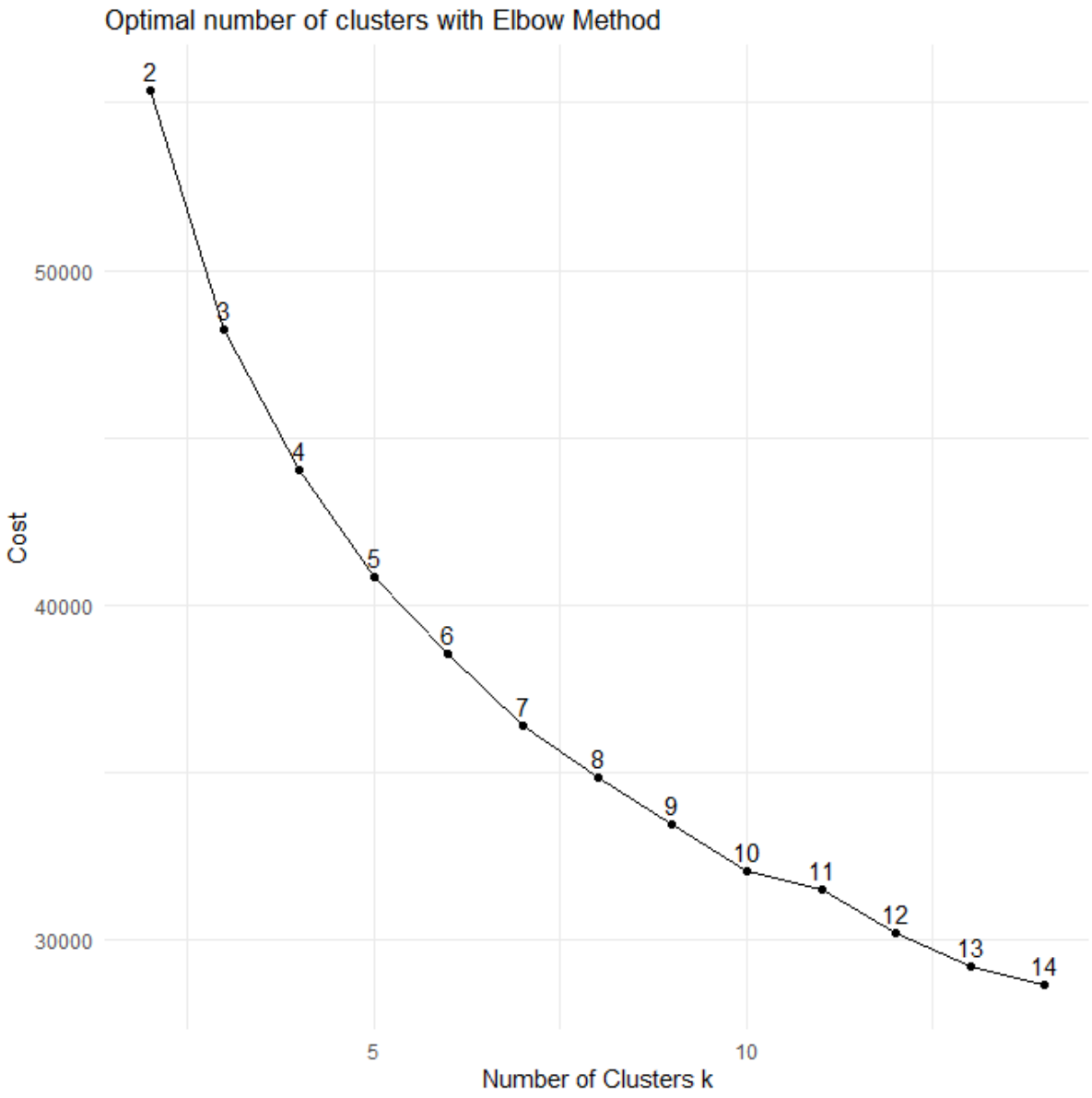
Elbow Method Setup

The Elbow Method is a commonly used technique to determine the optimal number of clusters in a dataset. The plot generated using this method displays the total within-cluster sum of squares (cost) against the number of clusters (k).

In the resulting graph, you will see a clear bend or "elbow" point. This point indicates the optimal number of clusters, balancing between too few clusters (resulting in high variance within each cluster) and too many clusters (potentially leading to overfitting).

For this specific dataset, the elbow appears to be around k = 5 or k = 6. This suggests that either 5 or 6 clusters would be a good choice for segmenting the data, as adding more clusters beyond this point yields diminishing returns in reducing the total within-cluster variance.

This balance ensures that the clusters are distinct enough to capture the underlying structure of the data without overcomplicating the model.



After determining the optimal number of clusters, the k-prototypes clustering algorithm was applied with k = 5. The k-prototypes algorithm is suitable for mixed-type data, handling both categorical and numerical variables effectively.

Cluster Summary and Visualization

The resulting clusters were analyzed, and a Multiple Correspondence Analysis (MCA) was performed to visualize the clustering of categorical variables in a 2-dimensional space. MCA is a technique used to explore relationships between categorical variables and is similar to Principal Component Analysis (PCA) but tailored for categorical data.

2D MCA Visualization

The 2D MCA plot provides a visual representation of the data points and their corresponding clusters:

- **Axes (Component 1 and Component 2)**: These represent the two most significant dimensions extracted from the categorical data, which best explain the variability in the data.
- **Points**: Each point in the plot represents a data observation.
- **Colors**: The color of each point indicates the cluster to which the observation belongs.

The plot helps in understanding how well the clusters are separated in the reduced-dimensional space. In this case, the 2D MCA visualization shows the distribution of data points across the five clusters, giving an intuitive understanding of the clustering results and the relationships between different categories within each cluster. This visualization is essential for verifying the effectiveness of the clustering and interpreting the characteristics of each cluster.

Outcome:

	Cluster	V1	V2
1	1	0.1389018	-0.2263285
2	2	0.3856426	-0.2456740
3	3	-0.9378236	-0.2300272
4	4	1.4503602	-0.1925271
5	5	-0.8106369	-0.2739554

where V1 is age and V2 is the balance.

Method 3: LLM + K-means

This hybrid approach involves using Language Models (LLMs) to enhance the clustering process, combining the strengths of LLMs in understanding textual data with the clustering capabilities of K-means.

1. Data Loading and Initial Inspection

Data is loaded from an Excel file containing the structured dataset. An initial inspection is performed to print and verify the column names, ensuring they match the expected structure required for subsequent processing.

2. Column Name Validation

A predefined list of expected column names is validated against the column names in the dataset. This step ensures that the dataset contains all necessary variables for the transformation process.

3. Data Cleaning

The dataset undergoes a cleaning process where rows with missing values (NAs) are removed. This ensures that the data is complete and ready for further processing.

4. Sentence Compilation

A custom function is defined to compile structured data into descriptive sentences. Each row of the dataset is transformed into a sentence that encapsulates the information in a readable format. This transformation facilitates the application of natural language processing techniques.

5. Text Vectorization and GloVe Model Fitting

The compiled sentences are tokenized and vectorized using the GloVe model. This involves several steps:

- **Tokenization**: Sentences are split into individual words (tokens).
- **Vocabulary Creation**: A vocabulary of unique tokens is built.
- **Term-Co-occurrence Matrix (TCM)**: A matrix representing the co-occurrence frequencies of tokens is created.
- **GloVe Model Training**: The GloVe model is trained on the TCM to generate word vectors, which capture the semantic relationships between words.

6. Sentence Embedding Creation

Each sentence is represented as an embedding by averaging the word vectors of its constituent tokens. This provides a fixed-length vector representation for each sentence, which can be used in various machine learning tasks. Sentences with no valid words are handled appropriately to avoid errors.

7. Outlier Detection and Removal

To ensure the quality of the embeddings, outliers are identified and removed:

- **Z-score Calculation**: Z-scores are computed for each feature in the embeddings to identify outliers.
- **Outlier Removal**: Embeddings with features exceeding a certain z-score threshold are removed from the dataset.

8. Saving Results

The cleaned embeddings are saved to a CSV file for future use. This allows for easy access and reuse in subsequent analyses or machine learning models.

9. Elbow Method for Determining Optimal Clusters

The script begins by performing the Elbow Method to identify the optimal number of clusters for k-means clustering. This involves running the k-means algorithm on the dataset for a

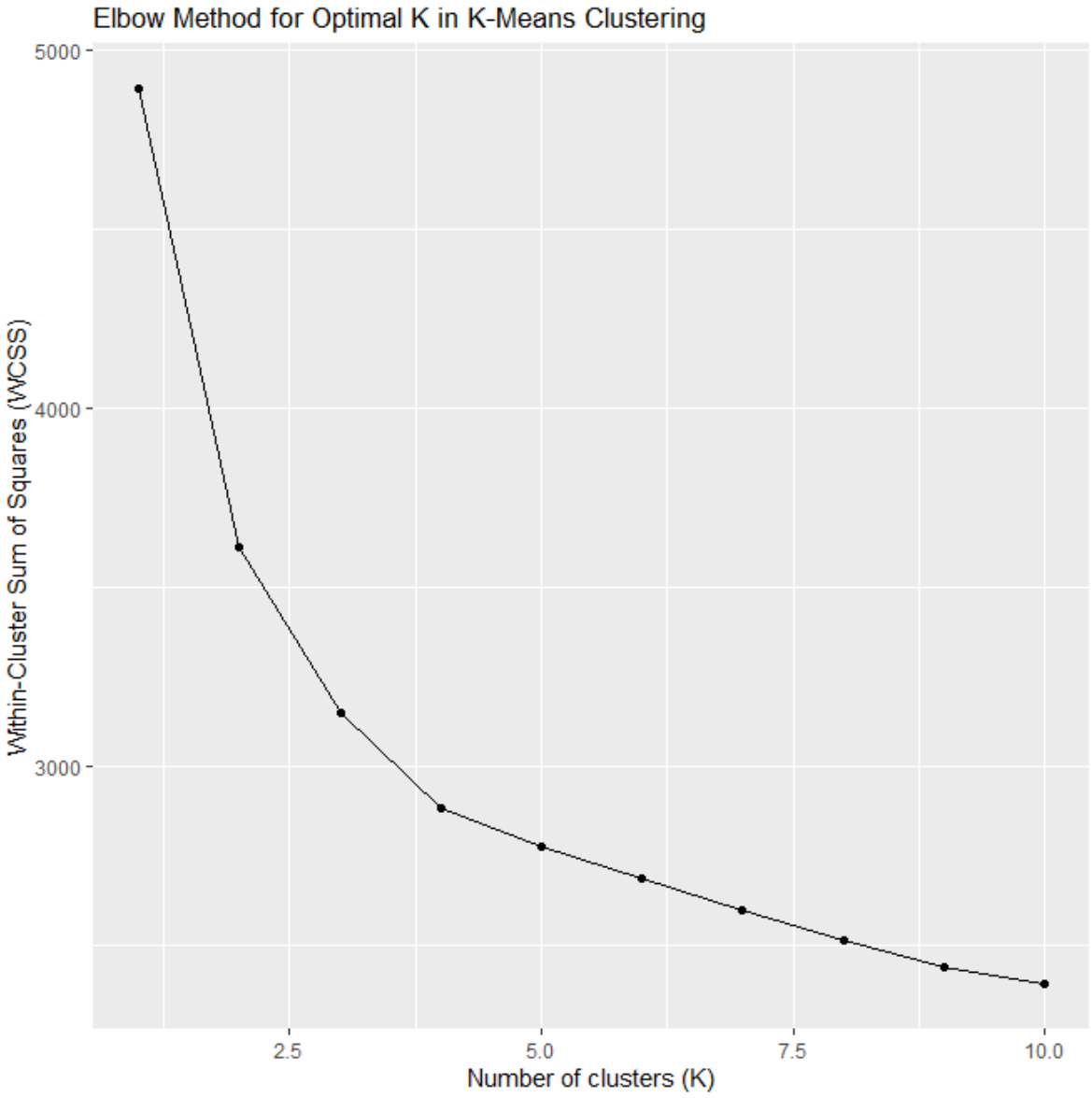
range of cluster numbers (k from 1 to 10) and calculating the Within-Cluster Sum of Squares (WCSS) for each k. The results are stored in a data frame.

10. Elbow Curve Plotting

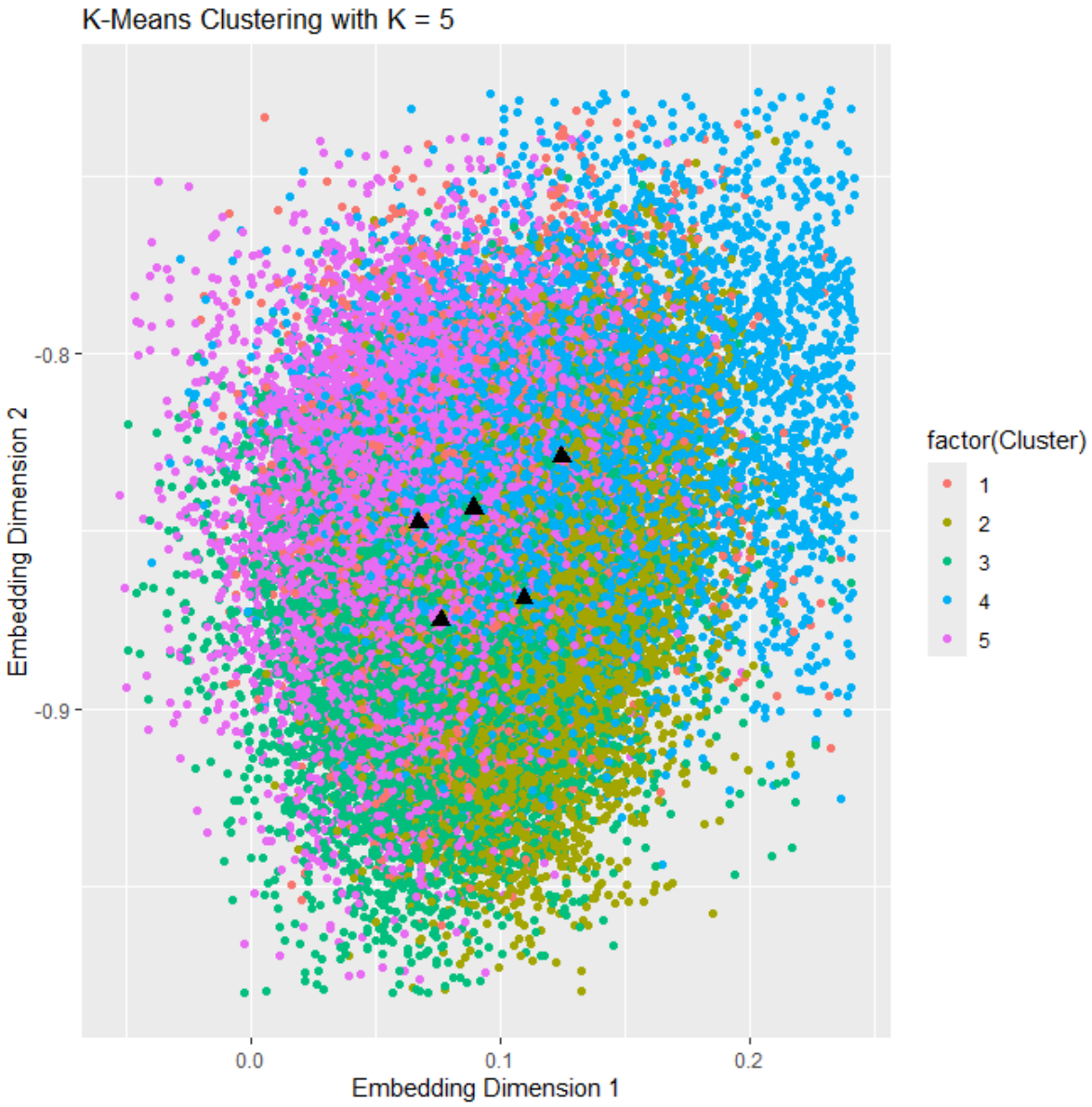
Using the ggplot2 library, an elbow curve is plotted to visualize the relationship between the number of clusters (k) and the WCSS. The plot helps identify the "elbow point," which indicates the optimal number of clusters. The x-axis represents the number of clusters, while the y-axis shows the WCSS. The plot includes a line connecting the points and labels for the axes and title.

11. K-Means Clustering with Optimal K

Based on the elbow method, k = 5 is chosen as the optimal number of clusters. The script then performs k-means clustering on the dataset with 5 clusters, initializing the algorithm 25 times to ensure robustness and stability of the results.



After performing k-means clustering with k=5k = 5k=5, the script extracts the cluster centers and prints them. These centers represent the mean position of all points within each cluster in the feature space.



2D PCA grapgh

4. Dimensionality Reduction Techniques

To visualize and interpret the clustering results, the following dimensionality reduction techniques are applied:

- **PCA (Principal Component Analysis)**: Reduces the dimensionality of the data while preserving as much variance as possible.
- **t-SNE (t-Distributed Stochastic Neighbor Embedding)**: Particularly useful for visualizing high-dimensional data.
- **MCA (Multiple Correspondence Analysis)**: Handles categorical data effectively.

5. Conclusion & Future work

In this research, we explored advanced techniques for customer segmentation, focusing on enhancing traditional clustering models with sophisticated approaches. Using a dataset from Kaggle's "Banking Dat aset — Marketing Targets," we implemented and compared three distinct methodologies: K-means, K-prototype, and a hybrid approach combining Language Models (LLMs) with K-means.

Each methodology was meticulously detailed, from preprocessing steps to outlier detection and cluster validation. The K-means algorithm demonstrated its robustness and versatility in handling numerical data, with the Elbow Method and Silhouette Analysis aiding in determining the optimal number of clusters. The K-prototype algorithm extended these capabilities to mixed data types, proving effective in segmenting customers based on both numerical and categorical variables.

Our hybrid approach, integrating LLMs with K-means, showcased the potential of leveraging textual data for clustering. By transforming structured data into descriptive sentences and utilizing the GloVe model for text vectorization, we enriched the clustering process, allowing for deeper insights and more meaningful segmentation.

Dimensionality reduction techniques such as PCA, t-SNE, and MCA were employed to visualize and interpret the clustering results, providing intuitive representations of high-dimensional data. These visualizations were crucial in verifying the effectiveness of the clustering and understanding the relationships within the data.

The project underscores the significance of thorough preprocessing, including outlier detection and removal, in enhancing model performance. Our findings reveal that the quality of data plays a critical role in achieving meaningful clusters. The hybrid LLM + K-means approach, in particular, highlights the importance of integrating advanced NLP techniques with traditional clustering methods to harness the full potential of the data.

Overall, this research contributes to the field of customer segmentation by offering comprehensive methodologies that combine traditional clustering algorithms with cutting-edge NLP techniques. Data scientists and analysts can adopt these methods to improve their clustering models, leading to more accurate and insightful customer segmentation. Future work could explore the integration of more sophisticated LLMs and the application of these techniques to other domains, further expanding the scope and impact of this research.

6. References

- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

- Aggarwal, C. C., & Reddy, C. K. (2014). *Data Clustering: Algorithms and Applications*. CRC Press.

- Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media.

- Huang, Z. (1997). Clustering large data sets with mixed numeric and categorical values. In *Proceedings of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 21-34).

- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

- Kaggle. (n.d.). *Banking Dataset — Marketing Targets*. Retrieved from Kaggle

- Scikit-learn documentation on clustering: Scikit-learn Clustering

- PyOD library documentation for outlier detection: PyOD Documentation

- Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining: KDD Conference