Enhancing Customer Segmentation Using Large Language Models (LLMs) and Deterministic, Independent-of-Corpus Embeddings (DICE)

M. N. S. Tissera
Department of Industrial Management
Univeristy of Kelaniya,
Kelaniya, Sri Lanka.
nimeshshamika@gmail.com

P. P. G. D. Asanka
Department of Industrial Management
Univeristy of Kelaniya,
Kelaniya, Sri Lanka.
dasanka@kln.ac.lk

R. A. C. P. Rajapakse
Department of Industrial Management
Univeristy of Kelaniya,
Kelaniya, Sri Lanka.
chathura@kln.ac.lk

2024 4th International Conference on Advanced Research in Computing (ICARC) | 979-8-3503-8486-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICARC61713.2024.10499784

***Abstract*—Customer segmentation is an instrumental aspect of marketing efforts that helps businesses understand and target their customers better. However, customer segmentation using traditional clustering methods faces challenges in dealing with diverse data types, capturing semantic information in textual features, and capturing minor differences between customers. This paper explores a novel approach that integrates Large Language Models (LLMs) and Deterministic, Independent-of-Corpus Embeddings (DICE) for customer segmentation. LLMs are used to generate sentence embeddings for textual features, while DICE is used to generate numeracy preserving word embeddings for numerical features. The embeddings are then combined and clustered using k-means. The paper compares the performance of the proposed approach with the conventional approach of clustering without embeddings on two real-world datasets from different domains. The results show that the hybrid approach achieves better cluster separation and evaluation metrics, demonstrating the potential of LLMs and DICE for enhancing customer segmentation.**

Keywords—Clustering, Customer Segmentation, Deterministic Independent-of-Corpus Embeddings (DICE), K-Means, Large Language Model (LLM)

I. Introduction

Customer segmentation is an essential aspect of marketing that helps businesses understand their customers better. It enables companies to tailor their marketing strategies to meet the specific needs of each segment, leading to increased customer satisfaction and loyalty [1], [2]. The importance of customer segmentation has increased in recent decades with the advent of one-customer strategies, especially in e-commerce. Traditional mass marketing in this area is becoming increasingly obsolete as customer-specific targeting becomes realizable [1].

The Segmentation, Targeting, and Positioning (STP) framework is a marketing model that consists of three steps: segmentation, targeting, and positioning. Segmentation is the process of dividing a heterogeneous market into smaller, more homogeneous groups based on similar characteristics such as demographics, psychographics, and behavior [2]. Targeting involves selecting one or more segments to focus on based on their attractiveness and compatibility with the company's objectives [2]. Positioning is the process of creating a unique image and identity for the product or service in the minds of the target customers [2].

Several methods have been proposed for customer segmentation, such as k-means, hierarchical, fuzzy, and model-based clustering [1]. One of the challenges of customer segmentation is how to represent the customer data in a way that preserves the semantic and contextual information. This is especially important when the customer data contains textual features, such as product reviews, feedback, or preferences. Traditional text representation methods, such as bag-of-words or TF-IDF, often fail to capture the meaning and nuances of natural language [3]. Therefore, there is a need for more advanced methods of text representation that can leverage the power of LLMs to generate sentence embeddings.

Most LLMs cannot generate embeddings for numerals such that their numeric properties are preserved. A solution is to use Deterministic, Independent-of-Corpus Embeddings (DICE) [4]. In this study, we will investigate using LLMs along with DICE for better customer segmentation.

The main contributions of this study are as follows:

- We compare the use of LLMs and DICE for customer segmentation to generate sentence embeddings for the customer features with the conventional approach of using them directly.
- We compare the performance of one of the most popular clustering techniques (i.e., k-means) used for customer segmentation with and without using LLMs and DICE for embedding creation on two real-world datasets from different domains.
- We investigate how the combination of LLMs and DICE can enhance the customer segmentation process.

II. Background and Related Work

A. Customer Segmentation: A Critical Aspect of Modern Marketing

Customer segmentation involves the categorization of a broad target market into subsets of consumers sharing common needs and priorities [5]. This enables businesses to direct their efforts toward specific customer groups rather than pursuing the entire market [5]. The significance of customer segmentation has grown notably in recent times, especially with the rise of personalized marketing strategies, rendering traditional mass marketing approaches outdated [1]. A profound understanding of individual customer interests and motivations is necessary, and effective customer segmentation proves instrumental in achieving this [1].

Machine learning has emerged as an effective tool for customer segmentation, allowing businesses to analyze extensive customer data and identify intricate patterns not readily discernible through manual analysis [6]. Machine learning algorithms facilitate segmentation based on factors like purchasing behavior, demographics, and psychographics [1], [6].

979-8-3503-8486-4/24/$31.00 ©2024 IEEE

Clustering, an unsupervised learning technique, stands out as one of the most prevalent machine learning methods for customer segmentation [1], [7], [8]. This method groups data points based on similarity or distance, revealing hidden patterns and structures within customer data and assigning labels to segments based on their characteristics [1], [7]. Out of the many clustering algorithms used for customer segmentation purposes, k-means is the most widely used method [1]. k-means is one of the simplest clustering algorithms, which partitions the data into K clusters based on the Euclidean distance between the data points and the cluster centroids [8].

In the realm of machine learning, selecting suitable features to describe customers becomes crucial. Various feature selection methods, including RFM analysis, PCA, MCA, Graph-based methods, and Purchase Trees, have been utilized [1]. Customer Lifetime Value (CLV) is commonly employed to identify profitable customers and formulate targeted strategies [9]. RFM analysis, a well-established method for measuring CLV, employs Recency (R), Frequency (F), and Monetary value (M) as key features to gauge customer behavior [10]. RFM analysis is particularly effective for e-commerce and retail businesses [11].

When considering previous literature on customer segmentation using machine learning, RFM analysis consistently emerges as a favored feature selection method [1]. Out of 105 publications reviewed, 44 utilized RFM analysis, citing its adaptability to diverse datasets and characteristics [1]. Manual selection, another prevalent feature selection method, involves expert-driven handcrafting of features, emphasizing the contextual appropriateness of the chosen method [1]. The choice between these methods depends on the specific context at hand [1].

B. *LLMs for Sentence Embedding Creation*

In recent years, there has been a growing interest in using Large Language Models (LLMs) for various Natural Language Processing (NLP) tasks, especially after the release of ChatGPT, a state-of-the-art LLM with 175 billion parameters that can perform various natural language tasks such as text classification, sentiment analysis, and language translation [12], [13]. LLMs are neural networks that process and generate natural language based on massive amounts of text data. [14].

Sentence embeddings are vector representations of sentences that capture their semantic meaning. They are useful for various natural language processing tasks, such as semantic textual similarity, text classification, and information retrieval [15].

The model 'paraphrase-multilingual-mpnet-base-v2' is a part of the Sentence Transformers library, designed to generate sentence embeddings. This model maps sentences and paragraphs to a 768-dimensional dense vector space and can be used for tasks like clustering or semantic search [16]. The model is based on the MPNet architecture. MPNet is a novel pre-training method that inherits the advantages of BERT and XLNet while avoiding their limitations [17]. It is designed to generate high-quality sentence embeddings, making it an excellent choice for our research. In terms of performance, the 'paraphrase-multilingual-mpnet-base-v2' model has an average performance score for sentence embeddings of 65.83 on encoding sentences over 14 diverse tasks from different domains [18]. It has an encoding speed of 2500 sentences/sec on a V100 GPU [18]. This indicates that the model provides a good balance between performance and computational efficiency. As for the size, the model has a size of approximately 278 million trainable parameters.

C. *DICE for Numeric Embedding Creation*

Despite the potential benefits the use of LLMs can bring to the table, there are some known drawbacks. One of the issues is the accuracy of generating sentence embeddings for numerical features [4]. The generated embedding for numerical features is that they may not reflect the original comparative characteristics, such as the magnitude. For example, the numbers 100 and 1000 have a significant difference in magnitude, but their sentence embeddings may not capture this difference. This may lead to inaccurate clustering results, as the numbers may be assigned to the same or different clusters based on their sentence embeddings rather than their actual values [4]. Therefore, generating sentence embeddings for numerical features may not preserve the ordinal and numerical relationships between the numbers. Hence, when generating embeddings, numerals are usually masked [4] so that they are not considered during embedding creation.

Masking numerals may not always be appropriate because, at times, these numerals may reflect important information; hence, masking them would cause a loss of this information. As a solution for this, we must generate embeddings for numerals such that their characteristics, such as the magnitude, are preserved. An approach proposed by [4] is to use Deterministic, Independent-of-Corpus Embeddings (DICE). Fig. 1 illustrates how DICE addresses the problem of preserving numerical characteristics. Embeddings for numbers ranging from 0 – 10,000 were created using the LLM and DICE. Then, they were plotted in a 3-D space using Principal Component Analysis (PCA). The visualization demonstrates that while the LLM could not create embeddings that preserve numeric properties, DICE was able to.

In this study, we will be incorporating DICE in addition to LLMs for embedding generation for numbers.

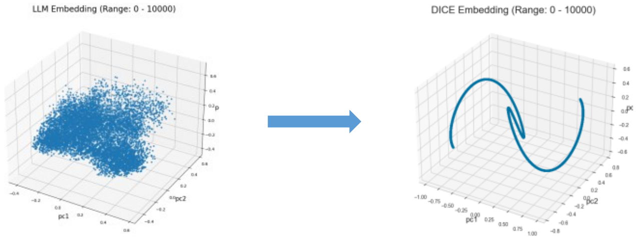


Fig. 1. The Problem Solved by DICE

III. Methodology

The methodology employed in this research involves a comparative study of customer segmentation using clustering techniques, specifically k-means. The comparison is conducted with and without integrating LLMs and DICE for sentence embedding creation. Hence, we conducted the experimentation in two phases. In the first phase, we segmented customers using k-means. In this phase, we did not generate embeddings for clustering. In the second phase, we used LLMs and DICE to generate embeddings for the customer features in the dataset. Finally, we compared the results of all the customer segmentation methods. The primary focus is on evaluating the performance of these clustering techniques on customer segmentation based on visualizations and on various metrics, including Silhouette score, Calinski score, and Davies Houldin score. The entire process was conducted on two distinct datasets from different domains to ensure the robustness and generalizability of the findings. TABLE I. describes the datasets. It is important to note that ground-truth clusters, or predefined clusters known in advance, are not available in the datasets under investigation.

TABLE I. DATASETS USED

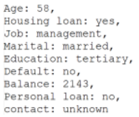
Dataset	Domain	Description
Banking Dataset - Marketing Targets [19] ***(Dataset – 1)***	Banking	This dataset contains information about customers who were contacted by a bank for a term deposit campaign.
E-Commerce Public Dataset by Olist [20] ***(Dataset – 2)***	E-Commerce	A collection of data from an online marketplace in Brazil. It comprises data from over 100,000 online orders in Brazil between 2016 and 2018.

A. Phase I (Without using LLMs)

In this phase, we used the derived/selected features of the dataset for clustering. For dataset-1, we selected features based on correlations. Two or more highly correlated features bring similar information to the model. Hence, removing one or more of these features does not compromise accuracy. To analyze correlations, we created a correlation matrix and considered two features to be highly correlated if their correlation coefficient is greater than or equal to 0.75. Furthermore, some features were irrelevant to our study (i.e., features that did not describe customers). Such features were also removed. Any categorical features were made numerical using one-hot encoding. After clustering, calculating the cluster performance metrics (i.e., Davies Bouldin Score, Calinski Score, and Silhouette Score) were based on the features we had selected for clustering.

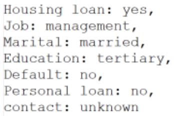
For dataset-2, the features were primarily numeric and not as descriptive as in dataset-1. Using existing features, we derived RFM values for each customer and segmented customers based on these RFM values. The reason for using RFM values was because it has been recognized as the most used feature selection method for customer segmentation as discussed in the “Related Work” section. Cluster evaluation metrics were calculated considering the RFM values (which were the features used for clustering).

B. Phase II (Using LLMs and DICE)



```
Age: 58,
Housing loan: yes,
Job: management,
Marital: married,
Education: tertiary,
Default: no,
Balance: 2143,
Personal loan: no,
contact: unknown
```

Fig. 2. Phase With Numerals



```
Housing loan: yes,
Job: management,
Marital: married,
Education: tertiary,
Default: no,
Personal loan: no,
contact: unknown
```

Fig. 3. Phase Without Numerals

During this phase, we generated embeddings for the features selected in Phase I. For dataset-1, we created phrases for each customer that contain both text and numbers and then created embeddings for these phrases. Embeddings were created using only the LLM. That is, we created a concatenated string for each customer that included both textual and numerical features as shown in Fig. 2 and then passed it to the LLM for embedding generation (this was done to compare the performance of this method of generating embeddings with that of incorporating DICE for numerals). Then, we clustered them. Next, we removed all numeric values from the phrases. For example, Fig. 3 shows a phrase without any numerical values after being removed from the phrase in Fig. 2. We then generated embeddings for the text and numerals in these phrases using the LLM and DICE, respectively. Textual features were concatenated to a single string and passed to the LLM. Embeddings for numerical features were generated by using DICE. These embeddings were then combined, as shown in Fig. 4.

In the case of dataset-2, all features used for clustering were purely numeric (because the features were based on RFM) unlike in dataset-1. Hence, as explained in the "Related Work" section, generating embeddings using an LLM does not make sense. Therefore, we used DICE for embedding generation. We then clustered these embeddings using k-means.

For dataset-1 and dataset-2, the cluster performance metrics were calculated based on the embeddings generated. In dataset-1, it was a vector space of 788 dimensions (768 + 20). In dataset-2, a vector space of 20 dimensions was used.

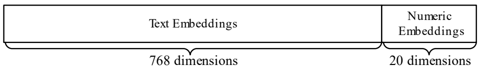


Fig. 4. Embedding Combination

IV. EXPERIMENTATION AND RESULTS

Since our study involves two phases (working with preprocessed data from the dataset and working with embeddings generated from the dataset), experimentation and results will be discussed separately for each.

A. Phase I

At this phase, we treated the dataset and clustered it as it is (i.e., no embeddings were created). Each dataset was preprocessed as required before clustering. Also, each dataset underwent clustering using k-means.

1) EDA and Preprocessing

Before performing any treatment to the datasets, we first conducted a thorough Exploratory Data Analysis (EDA). During the EDA phase, we focused on outlier detection, feature scaling, dimensionality reduction, correlation analysis, and data distribution.

The characteristics of each dataset are different; hence, the preprocessing techniques used for each are not the same. However, on a high level, the following are the generalized steps we followed when preparing the data for clustering.

1. Converting categorical variables into numeric ones.
2. Feature scaling to minimize feature bias.
3. Reducing dimensionality by removing one (or more) of the highly correlated features (since they bring similar information).
4. Detecting and removing outliers using Empirical Cumulative Distribution Functions (ECOD) (this is important for algorithms such as K-Means since it is sensitive to outliers [21])
5. Choosing the optimal number of clusters using the Elbow Method.

2) Clustering

Following the comprehensive preprocessing phase, each dataset was subjected to clustering using k-means. Literature mentions that k-means is the most used clustering method for customer segmentation [1].

K-means clustering is a partitioning-based algorithm that iteratively assigns data points to clusters until convergence. The number of clusters was determined using the Elbow Method. This method suggested a number of 4 clusters for both datasets.

For visualization purposes, we used Principal Component Analysis (PCA) to reduce the dimensionality. Fig. 5 shows the visualization of the data points on a 3-D (to the left) and 2-D (to the right) plane.

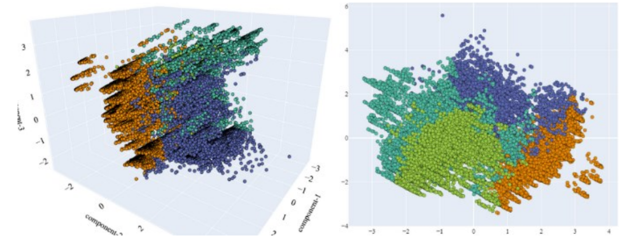


Fig. 5. K-Means for Dataset-1 (No Embeddings) – PCA

These plots show us that the clusters have almost no separation between them. Differentiating one cluster from the rest is a very challenging task, as the data points of one cluster overlap with those of another. However, we cannot immediately dive into conclusions since the cumulative variability of these three components of PCA is only 27.98%. This low value could be due to the complex polynomial relationships between features. We used the T-distributed Stochastic Neighbor Embedding (t-SNE) method to capture these complex relationships. The visualizations are in Fig. 6.

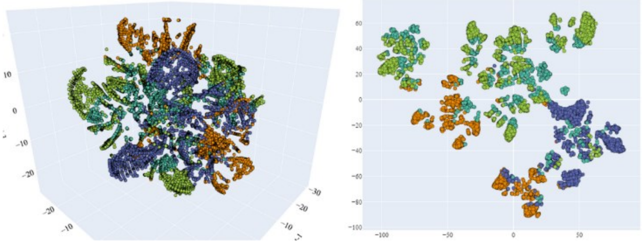


Fig. 6. K-Means for Dataset-1 (No Embeddings) – t-SNE

Even though the separation is clearer when using t-SNE, the results are not yet acceptable (i.e., clusters are not homogeneous enough). According to the visualizations and the results in TABLE II. , k-means and this dataset do not seem to go very well together.

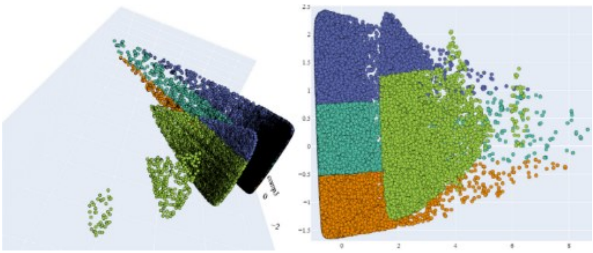


Fig. 7. K-Means for Dataset-2 (No Embeddings) – PCA

For dataset-2, the features that described the customers were mostly numerical. This information was not as descriptive as in dataset-1. It makes sense as well since this is a dataset taken from an e-commerce platform, and customers are not required (and will be reluctant to do so depending on the perceived relevance) to provide their details (except for information such as the email, name, and password) in most cases for such platforms. The data that can be captured from such platforms are primarily behavioral (i.e., their interaction with the platform). Hence, we decided to segment customers based on CLV. Specifically, we calculated the RFM values for each customer. As mentioned in the "Related Work" section, RFM-based segmentation is a popular customer segmentation method that attempts to cluster them based on their behavior.

Compared to dataset-1, cluster separation in this dataset seems to be better, as seen in Fig. 7. This is also indicated by the lower Davies-Bouldin Score (TABLE II.). Similarly, the dispersion within and between clusters is also higher, as indicated by the Calinski Score. For visualization on a 2-D and 3-D plane, we used PCA for feature selection. The cumulative variance for the three components selected using PCA was 100%. This makes sense because we have only 3 features. When we perform PCA to select 3 features (out of the 3 provided), the output should be the 3 features we provided as input (i.e., the recency, frequency, and monetary value).

TABLE II. CLUSTER EVALUATION METRICS FOR PHASE I

	Dataset – 1	Dataset – 2
Davies Bouldin Score	1.677	0.784
Calinski Score	6,914.705	76,187.329
Silhouette Score	0.167	0.399

B. *Phase II*

In this phase, we generated embeddings using an LLM and DICE for clustering and then compared their performance with the clusters created in Phase I.

1) *EDA and Preprocessing*

In the second phase of experimentation, initial EDA and preprocessing were not required since they were already done in Phase I. Embeddings were generated from the preprocessed data. When generating embeddings, we used both the LLM and DICE. As mentioned in the "Related Work" section, generating embeddings for numbers using an LLM does not make much sense. Hence, we used DICE for this purpose. After generating embeddings for the text and numerals separately, we combined them to represent a customer, as shown in Fig. 4.

2) *Clustering*

TABLE III. NUMBER OF CLUSTERS USING ELBOW METHOD

	Dataset – 1	Dataset – 2
No. of clusters (LLM)	4	-
No. of clusters (LLM & DICE)	4	-
No. of clusters (DICE)	-	4

The number of clusters was determined using the elbow method as done in Phase – I.

Interestingly, the number of clusters suggested by the elbow method for the generated embeddings was similar to those suggested during Phase – I (i.e., 4 clusters). A dash ('-') in TABLE III. indicates that the number of clusters was not determined for those respective cells since clustering was not applicable in those instances.

For comparison purposes, we clustered both the embedding datasets: the one generated only using the LLM and the one generated using the LLM and DICE for dataset-1.

This did not make sense for dataset-2 because features of this dataset used for segmentation were purely numeric. Hence, as discussed in the "Related Work" section, generating sentence embeddings for numbers using an LLM is senseless. Thus, we used DICE to generate embeddings for this dataset.

Fig. 8 is the visualization for clustering performed on the embedding generated using the LLM for dataset-1. PCA was used to select suitable dimensions for visualization. To generate these embeddings, we passed the customer features as text to the model (similar to Fig. 2)

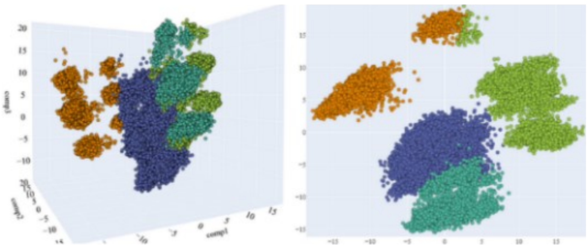


Fig. 8. K-Means for Dataset-1 (Embeddings from LLM) – PCA

Since the cumulative variance for the components selected using PCA were low values (<50%), we used the t-SNE method, hoping for better explanatory visualizations. Fig. 9 shows these. Surely, cluster separation has significantly improved with the use of t-SNE. We cannot expect well-formed clusters in this scenario since the evaluation metrics reflect the presence of cluster overlapping. While the overall dispersion between clusters is high (leading to a high Calinski-Harabasz score), the high Davies-Bouldin score, and low Silhouette score indicate the presence of cluster overlapping.

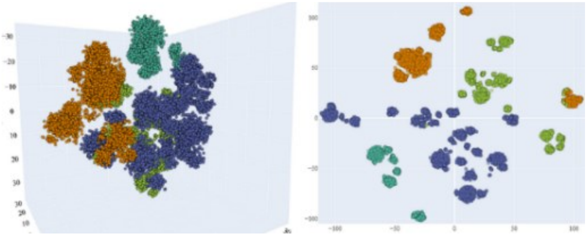


Fig. 9. K-Means for Dataset-1 (Embeddings from LLM) – t-SNE

Fig. 10 shows the visualization for the datasets generated using both the LLM and DICE. To generate embeddings for textual features using the LLM, we modified the input to the LLM. We did not pass any numeric features (age and balance) to the LLM. Instead, we generated embeddings for these numerical features using DICE and then combined the generated embeddings, as shown in Fig. 4. Like in the previous case, we used PCA for feature selection to plot the points in a 3-D and 2-D space. Since the cumulative variance for the components selected using PCA was high enough (>80%), implying that these visualizations do an excellent job of describing the cluster space, we did not have the need to use the more computationally expensive t-SNE method.

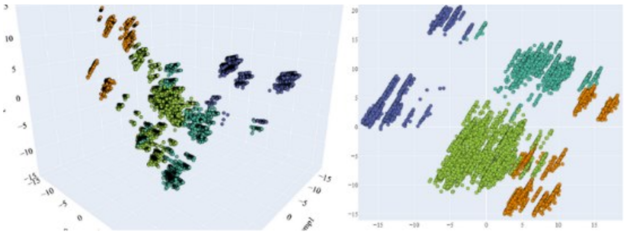


Fig. 10. K-Means for Dataset-1 (Embeddings from LLM and DICE) – PCA

The evaluation metrics demonstrate improved clustering. All three metrics used have shown considerable improvement. Moreover, rest assured that the numerical characteristics of numeric features (such as the magnitude) have been preserved. Hence, clustering using this method should appropriately consider all features for clustering.

Fig. 11 shows the clusters generated for dataset-2. We see that the quality of these clusters is quite high by looking at the visualization. Metrics in TABLE IV. justifies this. Since the cumulative variance for the components selected using PCA was high enough (>80%) for the components selected for visualization, t-SNE was not required. By comparing Fig. 11 with Fig. 7, since both seems to be somewhat similar, we can visually see how DICE preserves the numeric properties.

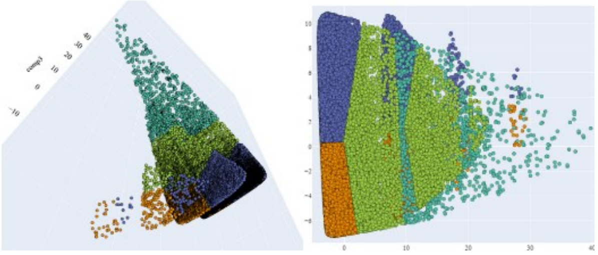


Fig. 11. K-Means for Dataset-2 (Embeddings from DICE) – PCA

TABLE IV. PHASE II CLUSTERING METRICS

	Dataset – 1		Dataset – 2
	LLM	LLM and DICE	DICE
Davies Bouldin Score	1.649	1.536	0.658
Calinski Score	7,069.150	8,808.201	79,777.919
Silhouette Score	0.258	0.378	0.501

V. DISCUSSION

Clustering is an unsupervised machine-learning technique. Validating the clustering results is not as straightforward as validating the results of supervised machine learning methods. Since the datasets we used were not labeled (i.e., customer clusters were unknown), we had to use internal cluster evaluation metrics. These metrics assess the quality of the clusters based on the data points within the clusters without using any external information or labels.

The results from Phase I, where clustering techniques were applied directly to preprocessed data without using LLMs, reveal interesting nuances in the performance of the methods across two distinct datasets. For dataset-1, characterized by a mix of categorical and numerical features, the k-means algorithm exhibited challenges in achieving well-defined cluster separation. The PCA visualization illustrated a significant overlap among clusters, indicating the limitations of k-means in this scenario. However, leveraging t-SNE brought about improved separation, albeit with indications of cluster overlapping.

On the other hand, dataset-2, primarily comprised of numerical features, demonstrated better cluster separation with k-means, corroborated by lower Davies-Bouldin scores and higher Calinski scores. This aligns with the notion that certain clustering methods might be more effective for datasets with specific characteristics, emphasizing the importance of tailoring approaches to the nature of the data.

Moving to Phase II, the integration of LLMs for sentence embedding creation presented a distinct layer of complexity to the customer segmentation task. For dataset-1, the use of LLM-generated embeddings showcased improved visual separation of clusters, particularly when employing t-SNE. Despite enhancements in visual clarity, evaluation metrics suggested the persistence of overlapping clusters, indicating the need for further refinement.

Interestingly, combining LLM-generated embeddings with embeddings from Deterministic, Independent-of-Corpus Embeddings (DICE) for numerical features demonstrated an enhanced clustering outcome with a significant improvement in clustering metrics. This integration addressed the challenge of preserving the characteristics of numeric features during embedding generation. The results suggest that a hybrid approach, incorporating LLMs for textual features and specialized methods for numerical features, contributes to more comprehensive and accurate customer segmentation.

The challenges observed in the direct clustering of numerical and categorical features highlight the need for adaptive clustering methods that can accommodate diverse data types. The integration of LLMs, while introducing complexities, presents an avenue for capturing subtle relationships within textual data, thereby enriching customer segmentation.

Moreover, the improved performance observed in Phase II, particularly with the hybrid approach, underscores the potential of combining the strengths of different methods for a more holistic understanding of customer behavior. This can be instrumental in personalized marketing, targeted recommendations, and enhanced user experiences.

VI. Conclusions and Future Work

This paper explores a novel approach for customer segmentation using Large Language Models (LLMs) and Deterministic, Independent-of-Corpus Embeddings (DICE). The proposed approach integrated LLMs for generating sentence embeddings for textual features and DICE for generating numeracy preserving word embeddings for numerical features. The embeddings were then combined and clustered using k-means. The results showed that the hybrid approach achieved better cluster separation and evaluation metrics, demonstrating the potential of LLMs and DICE for enhancing customer segmentation. Implications include the potential use of LLMs and DICE for personalized marketing, targeted recommendations, and enhanced user experiences.

This study opened up new avenues for customer segmentation using clustering with and without using embedding. The study can be further extended by focusing on aspects such as using more datasets, different clustering techniques, various LLMs, and alternative numerical embedding methods.

References

[1] M. Alves Gomes and T. Meisen, "A review on customer segmentation methods for personalized customer targeting in e-commerce use cases," *Inf Syst E-Bus Manage*, Jun. 2023, doi: 10.1007/s10257-023-00640-4.

[2] P. Kotler and K. L. Keller, *Marketing management*, 14th [ed.]. Upper Saddle River, N.J: Prentice Hall, 2012.

[3] M. N. S. Tissera and P. P. G. D. Asanka, "Optimizing Customer Segmentation: Comparing Customer Segmentation Techniques." [Online]. Available: https://drive.google.com/file/d/1K0MA2ZLuDIvYXlZHQJDTxYDDVv5G8FFB/view?usp=sharing

[4] D. Sundararaman, S. Si, V. Subramanian, G. Wang, D. Hazarika, and L. Carin, "Methods for Numeracy-Preserving Word Embeddings," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 4742–4753. doi: 10.18653/v1/2020.emnlp-main.384.

[5] J. N. Sari, L. E. Nugroho, R. Ferdiana, and P. I. Santosa, "Review on Customer Segmentation Technique on Ecommerce," *adv sci lett*, vol. 22, no. 10, pp. 3018–3022, Oct. 2016, doi: 10.1166/asl.2016.7985.

[6] A. Ranjan and S. Srivastava, "Customer segmentation using machine learning: A literature review," *AIP Conference Proceedings*, vol. 2481, no. 1, p. 020036, Nov. 2022, doi: 10.1063/5.0103946.

[7] R. Gupta, T. Jain, A. Sinha, and V. Tanwar, "Review on Customer Segmentation Methods Using Machine Learning," in *International Conference on IoT, Intelligent Computing and Security*, vol. 982, R. Agrawal, P. Mitra, A. Pal, and M. Sharma Gaur, Eds., in Lecture Notes in Electrical Engineering, vol. 982. , Singapore: Springer Nature Singapore, 2023, pp. 397–411. doi: 10.1007/978-981-19-8136-4_33.

[8] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, "K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data," *Information Sciences*, vol. 622, pp. 178–210, Apr. 2023, doi: 10.1016/j.ins.2022.11.139.

[9] C.-H. Cheng and Y.-S. Chen, "Classifying the segmentation of customer value via RFM model and RS theory," *Expert Systems with Applications*, vol. 36, no. 3, pp. 4176–4184, Apr. 2009, doi: 10.1016/j.eswa.2008.04.003.

[10] A. J. Christy, A. Umamakeswari, L. Priyatharsini, and A. Neyaa, "RFM ranking – An effective approach to customer segmentation," *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 10, pp. 1251–1257, Dec. 2021, doi: 10.1016/j.jksuci.2018.09.004.

[11] R. C. Blattberg, P. Kim, and S. A. Neslin, *Database marketing: analyzing and managing customers*. in International series in quantitative marketing, no. 18. New York: Springer, 2008.

[12] L. Fan, L. Li, Z. Ma, S. Lee, H. Yu, and L. Hemphill, "A Bibliometric Review of Large Language Models Research from 2017 to 2023".

[13] P. P. Ray, "ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope," *Internet of Things and Cyber-Physical Systems*, vol. 3, pp. 121–154, 2023, doi: 10.1016/j.iotcps.2023.04.003.

[14] M. U. Hadi *et al.*, "Large Language Models: A Comprehensive Survey of its Applications, Challenges, Limitations, and Future Prospects," preprint, Sep. 2023. doi: 10.36227/techrxiv.23589741.v3.

[15] T. Jiang, S. Huang, Z. Luan, D. Wang, and F. Zhuang, "Scaling Sentence Embeddings with Large Language Models." arXiv, Jul. 31, 2023. Accessed: Nov. 19, 2023. [Online]. Available: http://arxiv.org/abs/2307.16645

[16] "sentence-transformers/paraphrase-multilingual-mpnet-base-v2 · Hugging Face." Accessed: Nov. 19, 2023. [Online]. Available: https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2

[17] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "MPNet: Masked and Permuted Pre-training for Language Understanding." arXiv, Nov. 02, 2020. Accessed: Nov. 19, 2023. [Online]. Available: http://arxiv.org/abs/2004.09297

[18] "Pretrained Models — Sentence-Transformers documentation." Accessed: Nov. 19, 2023. [Online]. Available: https://www.sbert.net/docs/pretrained_models.html

[19] "Banking Dataset - Marketing Targets." Accessed: Nov. 29, 2023. [Online]. Available: https://www.kaggle.com/datasets/prakharrathi25/banking-dataset-marketing-targets

[20] Olist, "Brazilian E-Commerce Public Dataset by Olist." Kaggle. Accessed: Oct. 03, 2023. [Online]. Available: https://doi.org/10.34740/KAGGLE/DSV/195341

[21] X. Jin and J. Han, "K-Medoids Clustering," in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds., Boston, MA: Springer US, 2010, pp. 564–565. doi: 10.1007/978-0-387-30164-8_426.