MIMO Multipath Component Clustering using k-Deep Autoencoder

Casey H. Atienza[1], Sean David R. Aggabao[1], Thrisha Mae T. Banguis[1], Larie Joseph R. Lacsina[1], Vianca D. Manalo[1], Rhiz John P. Susi[1], Emmanuel T. Trinidad[1], Lawrence Materum[2,3, *]

[1] Department of Electronics Engineering, Don Honorio Ventura State University, Bacolor, Pampanga, Philippines
[2] Department of Electronics and Computer Engineering, De La Salle University, Manila, Philippines
[3] International Centre, Tokyo City University, Tokyo, Japan

ARTICLE INFO

ABSTRACT

In modern wireless communication systems, accurate characterization of wireless propagation channels remains a significant challenge. In multiple-input multiple-output (MIMO) wireless systems, a double-directional channel can be achieved by utilizing multipath components' spatial and temporal properties (MPCs). Grouping the MPC can simplify the parameters as a trade-off between complexity and accuracy. This paper implements k-Deep Auto Encoder (k-DAE), AE+k-means, and K Power Means (KPM) clustering approaches and compares their performance in clustering wireless propagation multipaths in indoor and outdoor scenarios. The results show that AE+k-means performs better than k-DAE in indoor scenarios by 25.48%, while k-DAE performs 24.60 % better in outdoor scenarios. The KPM algorithm performs best in all indoor scenarios among the three algorithms, with a significant increase of 4.38% and 11.062% to AE+k-means and k-DAE, respectively. However, both k-DAE and AE+k-means have quite similar performance in outdoor scenarios. The study also highlights the first use of autoencoders in clustering the MPCs. The results indicate that k-DAE can be used as an alternative clustering method in channel modeling. Future works envisioned applying the approach to other wireless channel models.

1. Introduction

Over the past few decades, wireless communication systems have experienced significant growth in capacity, accessibility, and applications by leveraging MIMO antenna systems. The development of Fifth-generation (5G) mobile technology relies on the deployment of massive MIMO systems [1] and the development of characterization of antennas for wearable systems [2]. Aside from the antenna configuration and design, one of the critical factors for next-generation systems is the wireless channel model which is a prerequisite in developing and evaluating system performance [3]. However, the accurate and efficient characterization of the propagation channel remains one of the challenges in wireless communications [4]. In wireless channels, Electromagnetic Waves (EM) that propagate along different paths interact with objects and suffer from reflection, scattering, and

---

* *Corresponding author.*
E-mail address: materuml@dlsu.edu.ph

diffraction. These paths are referred to as Multipath Components (MPC) and are distinguished by the estimated features, such as the delay, Angle of Arrival (AoA), Angle of Departure (AoD), and power. By grouping these spatial and temporal properties, the complexity of modeling the wireless channel can be reduced without sacrificing accuracy. Cluster-based channel models gain attraction due to their accuracy and minimal complexity. Among these models is the COST 2100 Channel Model (C2CM) [5], a geometry-based stochastic channel model that can be. Furthermore, by leveraging machine learning techniques, the automatic clustering of the MPCs can be realized, aiding the channel modeling process. Automatic clustering of MPC using algorithms started in the seminal works of Czink *et al.,* in previous works [6,7]. The framework is capable of clustering MPCs, deciding how many clusters to use, and discarding outliers was introduced. The study utilized the KPM algorithm, 3GPP channel model, and real-world MIMO datasets. Clustering algorithms have been proposed throughout the literature to cluster real-world and synthetic datasets [8-14]. Deep learning techniques have gained attention [15,16] and have been applied to several techniques, such as an autoencoder. An autoencoder is an unsupervised learning scheme under dimensionality reduction techniques. Its primary purpose is to generate accurate data representations (encoding) by training phase to avoid signal noise and reconstruct the input as the output. Autoencoders have been utilized to re-generate inputs into lower dimensions and represent the data in minimal parameters. Autoencoders have been utilized for data compression, sparse data representation, and anomaly detection [17].

Recently, the use of an autoencoder has been applied to the clustering problem because it can learn the manifold nature of data. Furthermore, the latent representation of the data in the embedded space of the autoencoder can be utilized for clustering or a joint optimization of clustering and encoding. Zheng *et al.,* [18] highlighted autoencoders with the purpose of improving the traditional method of cell scene division. The traditional method of cell scene division provides inaccuracy as well as it has no display or visualization, hence the proposal to create a new method of cell scene division with the use of an autoencoder and the k-means algorithm. The clustering function of an autoencoder was emphasized by Opochinsky *et al.,* [19], where an extended k-means algorithm to create a new deep clustering algorithm with the use of an autoencoder. The study aims to simplify the work of the clustering algorithm by representing each cluster with an autoencoder, instead of the standard centroid, that reconstructs data that belong to the same cluster. This approach can allow for more minor errors in reconstruction as well as avoidance of data collapsing, which is a known issue in using the traditional deep k-means algorithm. Data collapsing, in deep k-means clustering, happens when the vectors are collapsed at one point in the embedded space, and a single entity is formed from the centroids collapsing. The process of application is initiated with the training of the single autoencoder for the entire dataset. Several image datasets were used for the study, which include the Modified National Institute of Standards and Technology (MNIST) and Fashion datasets [20] that both have 70,000 images and the United States Postal Service (USPS) handwritten digit database. After training, the k-means is applied and is used to initialize the network parameters. Once network parameters are initialized, clustering is performed. The metrics for validation and evaluation include NMI, ARI, and ACC. After the procedure, the measures are taken at each dataset and compared with existing autoencoder algorithms such as Deep Autoencoder Mixture Clustering (DAMIC) [16], Deep Clustering Network (DCN) [21], and Deep Embedding Clustering (DEC) [22]. This paper presents the clustering of MPCs using the k-Deep AutoEncoder (k-DAE), a k-means algorithm mixed with an autoencoder, and measure the accuracy using external Clustering Validation Indices (CVI) and compares the clustering results with the well-known KPM algorithm. The first part of this paper introduces the cluster-based channel models in wireless communication in MIMO and the utilization of autoencoders in clustering tasks. The second part presents the methodology with the

algorithm used and the validity indices followed by the results and comparative analysis in section 3 and section 4 concludes this work.

2. Methodology

The methodology of the study is discussed in this section illustrated in Figure 1. The datasets consist of the indoor and outdoor scenarios extracted from the IEEE data port [21], which has eight scenarios and 30 generated snapshots. The datasets are fed to the k-Autoencoder to cluster multipaths using k number of autoencoders. The autoencoder initial clustering was also extracted, and the single autoencoder was utilized, followed by the k-means. External clustering metrics are employed to validate the results. Finally, the well-known KPM and the autoencoders Jaccard performances are compared.
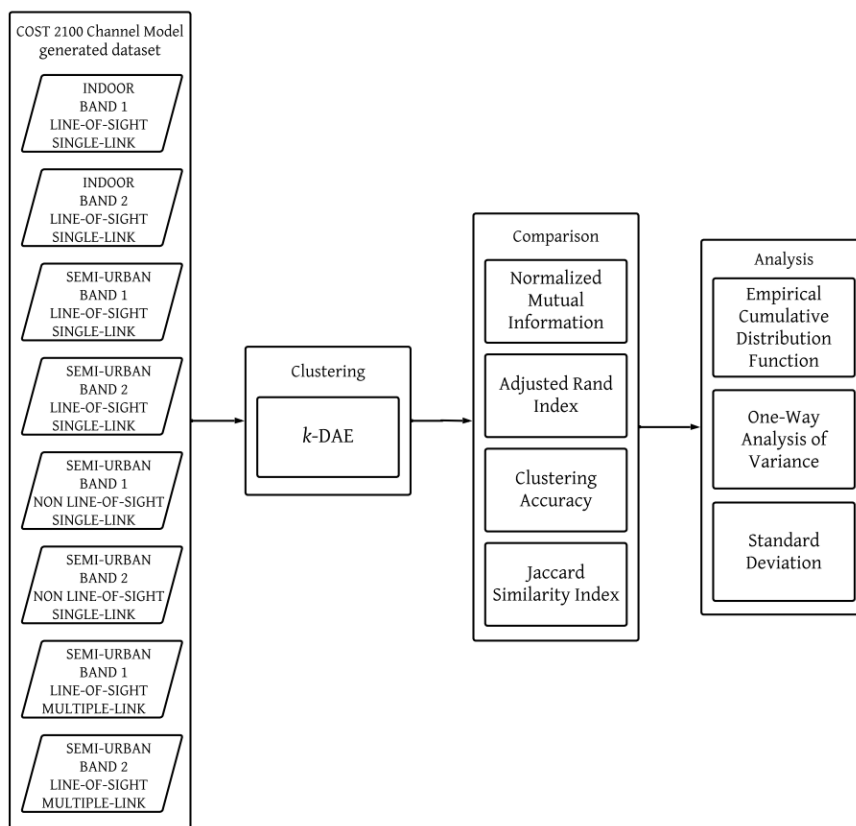

Fig. 1. Methodology

2.1 k-DAE Algorithm

The k-DAE algorithm is an extension of the well-known k-means algorithm where the cluster is represented by a k number of autoencoders [19]. The first step is the initialization of a single autoencoder to learn the input dataset. It also applies k-means to the embedded data, where the C_{initial} the output of the first iteration. k-DAE proceeds to train k autoencoders and aims to reconstruct the input data. The final computation C_{calc} is done by the hard clustering method of the k-means. The pseudocode of the k-DAE is summarized in Algorithm 1.

Algorithm 1: k-DAE Algorithm for C2CM clustering

Input(s):

$\boldsymbol{X}$ - Dataset excluding power and cluster IDs

$\boldsymbol{C}_{\text{ref}}$ – Reference Cluster ID

Output(s):

$\boldsymbol{C}_{\text{initial}}$ - Initial clustering results of AE+k means

$\boldsymbol{C}_{\text{calc}}$ – Cluster ID results of the k-DAE

Require: number of clusters k

i.   Initial training of a single AE based on k clusters

ii.   Apply k-means clustering on the embedded space

iii.   Set the number of AEs based on the number of clusters after the initial clustering

iv.   Each cluster represents an AE instead of a centroid

$$x \rightarrow \hat{x}(i) = f_i(x; \theta_i), i = 1, \ldots, k$$

v.   Training: Clustering of each AE with the initial clustering and minimize the reconstruction error

$$L(\theta_1, \ldots, \theta_k) = \sum_{t=1}^{n} \min_i d(x_t, \hat{x}_t(i))$$

vi.   Set the number of AEs based on the number of clusters after the 2nd training of autoencoders

vii.   Final Clustering; computation of $\boldsymbol{C}_{\text{calc}}$ or the reconstruction of input

$$\hat{c}_t = \arg \min_{i=1}^{k} d(x_t, \hat{x}_t(i)), t = 1, \ldots, n$$

2.2 Cluster Validity Indices

After obtaining the clustering results C_{calc}, an ensemble of validity indices is used to compare the results to the ground-truth labels C_{ref}, thereby assessing the performance of the algorithm to the C2CM dataset. Due to the availability of ground truth, external CVIs are used, namely, NMI, Accuracy [23], ARI [24], and the Jaccard index [25,26]. The normalized mutual index is a measure of similarity used for contrasting community detection approaches. Mathematically, the NMI is given by Eq. (1):

$$\text{NMI}(\boldsymbol{C}_{\text{ref}}, \boldsymbol{C}_{\text{calc}}) = \frac{MI(\boldsymbol{C}_{\text{ref}}, \boldsymbol{C}_{\text{calc}})}{\max[H(\boldsymbol{C}_{\text{ref}}), H(\boldsymbol{C}_{\text{calc}})]} \tag{1}$$

where MI is the mutual information, and H indicates the entropy. The value ranges from [0,1]. Furthermore, the Adjusted Rand Index (ARI), considers the number of instances that occur within the same cluster and those that exist in distinct clusters. ARI is defined in Eq. (2).

$$\text{ARI} = \frac{M_{11} + M_{00}}{M_{00} + M_{01} + M_{10} + M_{11}} = \frac{M_{11} + M_{00}}{\binom{M}{2}} \tag{2}$$

where M_{11} is the number of pairs that are part of the same cluster, M_{00} is the number of instances that are part of different clusters. M_{01} is the number of pairs that are in the reference cluster but not in the clustering output and M_{10} represents the pairs that are in C_{calc} but not in C_{ref}.

The clustering accuracy (ACC) which evaluates the proportion of data points for which the produced clusters can be successfully mapped to the ground truth classes given by Eq. (3) where N is the total number of data, $\delta(x, y)$ is the delta function that equates to 1 if $x = y$ and 0 otherwise. The map(C_{calc}) is the permutation mapping function.

$$\text{ACC} = \frac{\sum_{i=1}^{N} \delta(C_{\text{ref}}, \text{map}(C_{\text{calc}}))}{N} \tag{3}$$

The Jaccard index is a measure of similarity between C_{calc} and C_{ref} and is the intersection over the union. The Jaccard index tends to be sensitive to the number of clusters. In multipath clustering, the Jaccard index is used to validate the similarity of the clustering solution to the ground truth [25]. The Jaccard index can be computed using Eq. (4).

$$\eta_{\text{Jac}} = \frac{|C_{\text{ref}} \cap C_{\text{cal}}|}{|C_{\text{ref}} \cup C_{\text{cal}}|} = \frac{M_{11}}{M_{11} + M_{10} + M_{01}} \tag{4}$$

3. Results

The performance of the algorithm to the C2CM dataset is examined with these CVIs. The initial clustering was also extracted and compared to the k-DAE. Table 1 shows the CVI scores per scenario, thus comparing the two algorithms, AE+k-means, and k-DAE. The AE+k-means algorithm has the highest mean values in the indoor scenarios. The highest mean shows a value of 0.8930 under the NMI index, 0.6720 under the ARI index, 0.7890 under the ACC index, and 0.5284 under the Jaccard index, all on the Indoor B1 scenario, proving that the initial training shows favorable results in the indoor scenario. On the other hand, the acquired data k-DAE algorithm presents the highest mean values in the outdoor scenarios that extend up to 0.4237 in terms of its NMI index under the Semi-Urban B1 ML LOS, 0.1091 in terms of its ARI index under the semi-urban B1 multiple-link, 0.2424 in terms of its ACC index under the semi-urban single link LOS, and 0.0531 in terms of its Jaccard index also under the Semi-Urban SL LOS, proving that the deep training of data provides good results with regards to the outdoor scenarios.

Table 1
Mean CVI scores per Scenario

Scenarios	CVI	AE+ k- means	k-DAE
Indoor B1 LOS Single Link	NMI	0.9148	0.7417
	ARI	0.7093	0.3131
	ACC	0.8153	0.5
	Jaccard	0.591	0.2391
Indoor B2 LOS Single Link	NMI	0.8996	0.6772
	ARI	0.6372	0.2684
	ACC	0.7646	0.4616
	Jaccard	0.4773	0.3052
Semi-Urban B1 LOS Single Link	NMI	0.3681	0.4063
	ARI	0.1015	0.0973
	ACC	0.2387	0.2373
	Jaccard	0.046	0.052
Semi-Urban B2 LOS Single Link	NMI	0.3724	0.3973
	ARI	0.0997	0.0989
	ACC	0.2276	0.2368
	Jaccard	0.0446	0.046
Semi-Urban B1 NLOS Single Link	NMI	0.2967	0.3706
	ARI	0.0925	0.096
	ACC	0.2056	0.2216
	Jaccard	0.0422	0.0486
Semi-Urban B2 NLOS Single Link	NMI	0.2925	0.3878
	ARI	0.0895	0.1107
	ACC	0.193	0.234
	Jaccard	0.0482	0.0476
Semi-Urban B1 LOS Multiple Links	NMI	0.3753	0.4282
	ARI	0.1061	0.1104
	ACC	0.2154	0.2411
	Jaccard	0.0336	0.0337
Semi-Urban B2 LOS Multiple Links	NMI	0.3916	0.414
	ARI	0.1162	0.1051
	ACC	0.2273	0.2379
	Jaccard	0.0291	0.0328

The analysis of variance and empirical cumulative distribution function (ECDF) is performed to analyze the Jaccard indices performance of the two algorithms and are compared to the KPM algorithm typically used in clustering the multipaths. Figure 2(a) shows the ANOVA of the indoor scenarios. The KPM presents the highest scores in the indoor scenarios with a median and 75th percentile of 1 and a 25th percentile of 0.6251. In Figure 2(b), among the three algorithm performances in the outdoor scenarios, k-DAE has the highest Jaccard score with a median of 0.0435, a 75th percentile of 0.0593, and a 25th percentile of 0.0307.
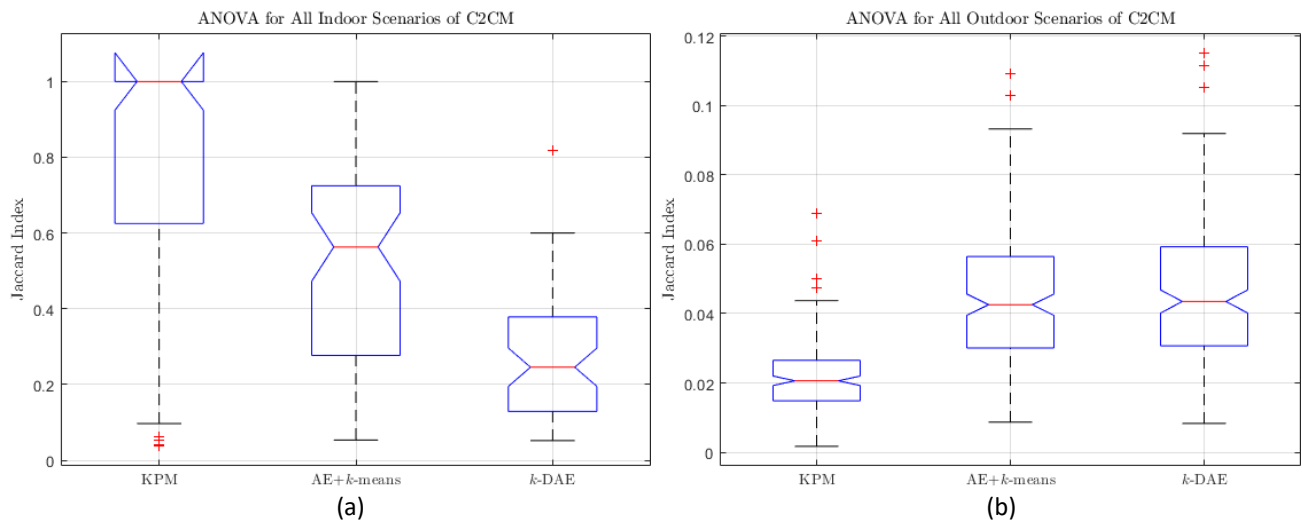
Fig. 2. ANOVA of Jaccard Indices for (a) Indoor Scenarios (b) Outdoor Scenarios

The ECDF of the Jaccard index of all indoor scenarios is illustrated in Figure 3(a). The ECDF of AE+k-means shows an increase from the KPM in the 10th percentile only. However, for the median and the 90th percentile, AE+k-means and k-DAE are outperformed by the KPM, which has a value of unity that is for perfect clustering performance. For the outdoor scenarios, the ECDF is shown in Figure 3(b). The performance of AE+k-means has significantly increased, with 0.0119, 0.022, and 0.0348 in the 10th, 50th, and 90th percentile, respectively. On the other hand, the k-DAE shows an increase of 0.0118, 0.0229, and 0.424 on the same percentiles.
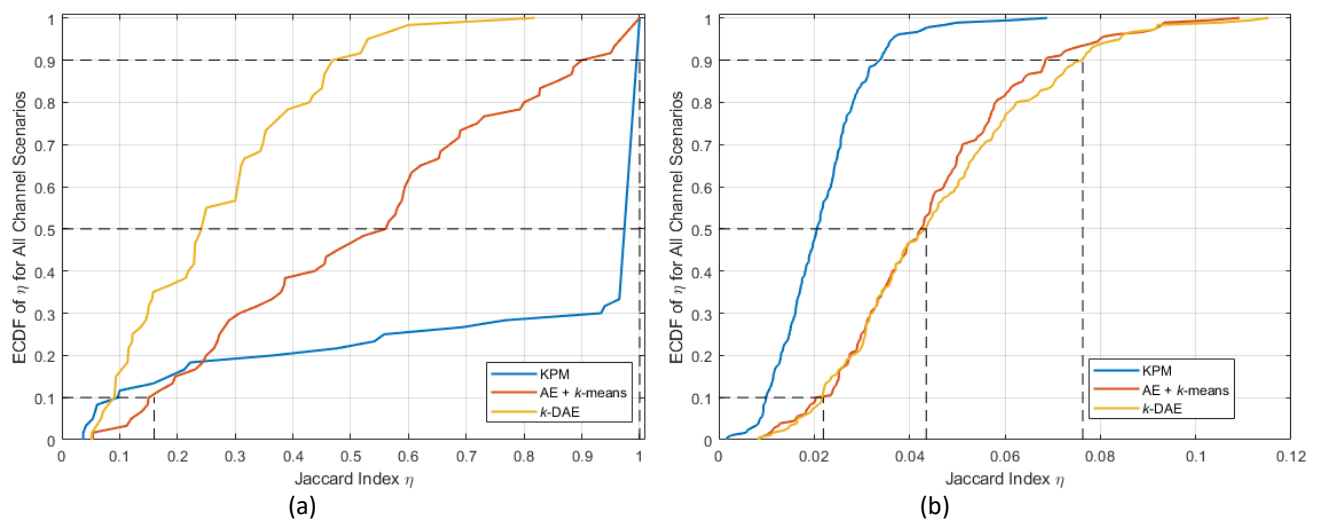

Fig. 3. ECDF of Jaccard Indices for (a) Indoor Scenarios (b) Outdoor Scenarios

Finally, the Jaccard index performances of all scenarios are compared using ECDF, illustrated in Figure 4. The k-DAE shows an increase of 0.0116 in the 10th and 0.0283 in the 50th percentile. This result is due to the indoor scenarios included, where KPM performs well in clustering the multipaths. The results show that the k-DAE have comparable performance ranging from 0 to 0.1 Jaccard index of the outdoor scenarios, which have a large number of multipaths. While for the indoor, KPM still has a value of 0.8 to 1 in the 90th percentile.
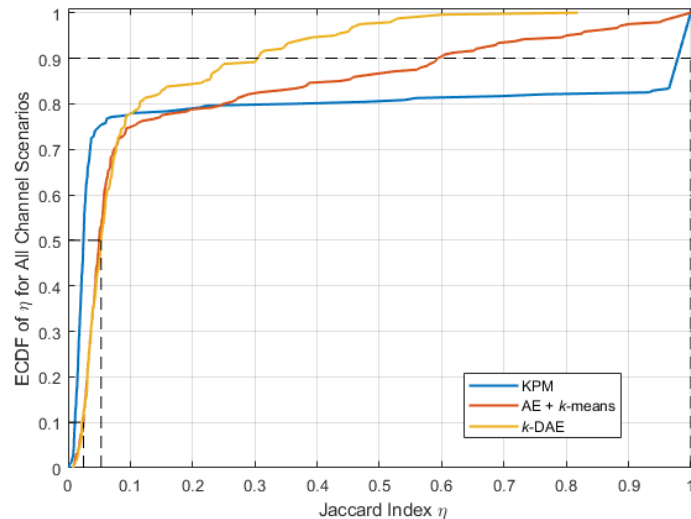
Fig. 4. ECDF of Jaccard Indices for all scenarios

A post-hoc Tukey Test with the Honestly Significant Difference (HSD) was also done for all scenarios and is summarized in Table 2. This test was done to see the significant differences between the Jaccard index means of the three algorithms. The post-hoc test reveals that the group means of KPM and AE+k-means are not significantly different, which is also true for the k-DAE and AE+k-means. Consequently, the group means of KPM and k-DAE, suggest a significant difference greater than the Tukey HSD value. This result is due to the higher number of outdoor scenarios that still remains a challenge to cluster a large number of MPCs. Hence, this work shows that KPM still performs better in clustering a low number of MPCs and is still a challenge in the outdoor scenario, where the k-DAE shows comparable performance.

Table 2
Post-Hoc Tukey Test for all scenarios

Algorithm		Mean Difference	Tukey HSD Value
KPM	AE+k-means	0.048376	< 0.07238
KPM	k-DAE	0.11062	> 0.07238
AE+k -means	k-DAE	0.062239	< 0.07238

This work utilizes a deep autoencoder approach in clustering multipath waves as an aid to model the double-directional properties of the wireless channel. The results also show that clustering higher number of multipaths and multiple-links scenarios is still challenging. Hence, the utilization of different clustering approaches is deemed necessary to improve the clustering leading to reduced parameters and increasing the accuracy of cluster-based MIMO channel models. Future work involves applying the algorithm to other standard channel models and investigating the application of tracking the MPCs such as in [27].

4. Conclusions

This paper implements the *k-DAE* in clustering the C2CM MPC dataset. The analysis of the performance of the *k-DAE, AE+k-means,* and *KPM* algorithms is presented. The autoencoders and deep autoencoders are utilized to cluster the MPCs. In conclusion, the validation and accuracy of clustering using autoencoders have a performance comparable to that of the KPM algorithm, particularly in outdoor scenarios. However, in indoor scenarios, KPM is still superior. The *k-DAE* algorithm increases by 1.18% in the 10th percentile, 2.29% in the 50th percentile, and 4.24% in the

90th percentile in all outdoor scenarios from the KPM. Thus, *k*-DAE can be utilized when dealing with outdoor parameters and can serve as an alternate clustering method for MIMO multipath waves. The use of different activation functions for the autoencoder is considered for future work.

Acknowledgement

References

[1] Galleto Jr, Fredelino A., Aaron Don Africa, and Lawrence Materum. "Impact of Circuit Power on the Energy Efficiency of Massive Multiple-Input Multiple-Output (MIMO) Cellular Systems." *Journal of Advanced Research in Applied Sciences and Engineering Technology* 32, no. 2 (2023): 164-174. https://doi.org/10.37934/araset.32.2.164174

[2] Abd Razak, Jeefferie, Nor Aisah Khalid, Hazman Hasib, Mazlin Aida Mahamood, Mohd Muzafar Ismail, Noraiham Mohamad, Poppy Puspitasari, and Moayad Husein Flaifel. "Electrical conductivity and antenna properties of polyaniline filled GNPs nanocomposites." *Malaysian Journal on Composites Science & Manufacturing* 4, no. 1 (2021). https://doi.org/10.37934/mjcsm.4.1.1127

[3] Molisch, Andreas F., and Fredrik Tufvesson. "Propagation channel models for next-generation wireless communications systems." *IEICE Transactions on Communications* 97, no. 10 (2014): 2022-2034. https://doi.org/10.1587/transcom.E97.B.2022

[4] Aldossari, Saud Mobark, and Kwang-Cheng Chen. "Machine learning for wireless communication channel modeling: An overview." *Wireless Personal Communications* 106 (2019): 41-70. https://doi.org/10.1007/s11277-019-06275-4

[5] Liu, Lingfeng, Claude Oestges, Juho Poutanen, Katsuyuki Haneda, Pertti Vainikainen, François Quitin, Fredrik Tufvesson, and Philippe De Doncker. "The COST 2100 MIMO channel model." *IEEE Wireless Communications* 19, no. 6 (2012): 92-99. https://doi.org/10.1109/MWC.2012.6393523

[6] Czink, Nicolai, Pierluigi Cera, Jari Salo, Ernst Bonek, Jukka-Pekka Nuutinen, and Juha Ylitalo. "A framework for automatic clustering of parametric MIMO channel data including path powers." In *IEEE Vehicular Technology Conference*, pp. 1-5. IEEE, 2006. https://doi.org/10.1109/VTCF.2006.35

[7] Czink, Nicolai, Pierluigi Cera, Jari Salo, Ernst Bonek, J-P. Nuutinen, and Juha Ylitalo. "Improving clustering performance using multipath component distance." *Electronics Letters* 42, no. 1 (2006): 33-35. https://doi.org/10.1049/el:20063917

[8] Materum, Lawrence, Jun-ichi Takada, Ichirou Ida, and Yasuyuki Oishi. "Mobile station spatio-temporal multipath clustering of an estimated wideband MIMO double-directional channel of a small urban 4.5 GHz macrocell." *EURASIP Journal on wireless communications and networking* 2009 (2009): 1-16. https://doi.org/10.1155/2009/804021

[9] Li, Yupeng, Jianhua Zhang, Zhanyu Ma, and Yu Zhang. "Clustering analysis in the wireless propagation channel with a variational Gaussian mixture model." *IEEE Transactions on Big Data* 6, no. 2 (2018): 223-232. https://doi.org/10.1109/TBDATA.2018.2840696

[10] Teologo, Antipas T., and Lawrence Materum. "Cluster-wise Jaccard accuracy of KPower means on multipath datasets." *International Journal of Emerging Trends in Engineering Research* 7, no. 8 (2019): 203. https://doi.org/10.30534/ijeter/2019/16782019

[11] Blanza, Jojo F., and Lawrence Materum. "Joint identification of the clustering and cardinality of wireless propagation multipaths." *International Journal of Emerging Trends in Engineering Research* 7, no. 12 (2019): 762. https://doi.org/10.30534/ijeter/2019/057122019

[12] Trinidad, Emmanuel T., and Lawrence Materum. "High-Accuracy Visualization-Based Grouping of MIMO Multipath Waves." *Journal of Communications* 18, no. 1 (2023). https://doi.org/10.12720/jcm.18.1.68-75

[13] Hanpinitsak, Panawit, Kentaro Saito, Junichi Takada, Minseok Kim, and Lawrence Materum. "Clustering method based on scatterer locations for indoor dynamic MIMO channel." In *2016 10th European Conference on Antennas and Propagation (EuCAP)*, pp. 1-4. IEEE, 2016. https://doi.org/10.1109/EuCAP.2016.7481586

[14] Du, Fei, Xiongwen Zhao, Yu Zhang, Yang Wen, Zihao Fu, Suiyan Geng, Peng Qin et al. "SVM-assisted adaptive kernel power density clustering algorithm for millimeter wave channels." *IEEE Transactions on Antennas and Propagation* 70, no. 6 (2022): 4014-4026. https://doi.org/10.1109/TAP.2022.3140538

[15] DeepAI, "Autoencoder," *DeepAI*. DeepAI, May 2019. [Online]. Available: https://deepai.org/machine-learning-glossary-and-terms/autoencoder

[16] Chazan, Shlomo E., Sharon Gannot, and Jacob Goldberger. "Deep clustering based on a mixture of autoencoders." In *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1-6. IEEE, 2019. https://doi.org/10.1109/MLSP.2019.8918720

[17] He, Ruisi, and Zhiguo Ding, eds. *Applications of machine learning in wireless communications*. Vol. 81. Telecommunications, 2019.

[18] Zeng, Jun, Juan Wang, Liang Guo, Guanghui Fan, Kaixuan Zhang, and Guan Gui. "Cell scene division and visualization based on autoencoder and k-means algorithm." *IEEE Access* 7 (2019): 165217-165225. https://doi.org/10.1109/ACCESS.2019.2953184

[19] Opochinsky, Yaniv, Shlomo E. Chazan, Sharon Gannot, and Jacob Goldberger. "K-autoencoders deep clustering." In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4037-4041. IEEE, 2020. https://doi.org/10.1109/ICASSP40776.2020.9053109

[20] Xiao, Han, Kashif Rasul, and Roland Vollgraf. "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms." *arXiv preprint arXiv:1708.07747* (2017).

[21] Yang, Bo, Xiao Fu, Nicholas D. Sidiropoulos, and Mingyi Hong. "Towards k-means-friendly spaces: Simultaneous deep learning and clustering." In *international conference on machine learning*, pp. 3861-3870. PMLR, 2017.

[22] Xie, Junyuan, Ross Girshick, and Ali Farhadi. "Unsupervised deep embedding for clustering analysis." In *International conference on machine learning*, pp. 478-487. PMLR, 2016.

[23] Cai, Deng, Xiaofei He, and Jiawei Han. "Locally consistent concept factorization for document clustering." *IEEE Transactions on Knowledge and Data Engineering* 23, no. 6 (2010): 902-913. https://doi.org/10.1109/TKDE.2010.165

[24] Fahad, Adil, Najlaa Alshatri, Zahir Tari, Abdullah Alamri, Ibrahim Khalil, Albert Y. Zomaya, Sebti Foufou, and Abdelaziz Bouras. "A survey of clustering algorithms for big data: Taxonomy and empirical analysis." *IEEE transactions on emerging topics in computing* 2, no. 3 (2014): 267-279. https://doi.org/10.1109/TETC.2014.2330519

[25] Teologo Jr, Antipas, Lawrence Materum, Jojo Blanza, and Takuichi Hirano. "Comparative study of k-power means, ant colony optimization, kernel power density-based estimation, and Gaussian mixture model for wireless propagation multipath clustering." *International Journal* 8, no. 7 (2020). https://doi.org/10.30534/ijeter/2020/164872020

[26] Rodriguez, Mayra Z., Cesar H. Comin, Dalcimar Casanova, Odemir M. Bruno, Diego R. Amancio, Luciano da F. Costa, and Francisco A. Rodrigues. "Clustering algorithms: A comparative approach." *PloS one* 14, no. 1 (2019): e0210236. https://doi.org/10.1371/journal.pone.0210236

[27] Hanpinitsak, Panawit, Kentaro Saito, Jun-ichi Takada, Minseok Kim, and Lawrence Materum. "Multipath clustering and cluster tracking for geometry-based stochastic channel modeling." *IEEE Transactions on Antennas and Propagation* 65, no. 11 (2017): 6015-6028. https://doi.org/10.1109/TAP.2017.2754417