Proceedings of CONF-MLA 2024 Workshop: Semantic Communication Based Complexity Scalable Image Transmission System for Resource Constrained Devices
DOI: 10.54254/2755-2721/83/2024GLG0060

An exploration of KANs and CKANs for more efficient deep learning architecture

Shuhui Zhou

DUT School of Software Technology & DUT-RU International School of Information Science and Engineering, Dalian University of Technology, Dalian, Liaoning, 116620, China

zshwan@mail.dlut.edu.cn

Abstract. Deep learning has revolutionized the field of machine learning with its ability to discern complex patterns from voluminous data. Despite the success of Multi-Layer Perceptrons (MLPs) and Convolutional Neural Networks (CNNs), there is an ongoing quest for architectures that offer higher expressiveness with fewer parameters. This paper focuses on the Kolmogorov-Arnold Networks (KANs) and Convolutional Kolmogorov-Arnold Networks (CKANs), which integrate learnable spline functions for enhanced expressiveness and efficiency. This study designs a range of networks to compare KANs with MLPs and CKANs with classical CNNs on the CIFAR-10 dataset. Moreover, this study evaluates the models based on several metrics, including accuracy, precision, recall, F1 score, and parameter count. Based on the experimental results, networks with KANs and CKANs demonstrated improved accuracy with a reduced parameter footprint, indicating the potential of KAN-based models in capturing complex patterns. In conclusion, integrating KANs into CNNs and MLPs is a promising approach for developing more efficient and interpretable models, offering a path to advance deep learning architectures.

Keywords: Kolmogorov-Arnold network, Neural network, Convolutional neural network.

1. Introduction

Deep learning is a machine learning method based on artificial neural networks and representation learning. Deep learning is used to enable computers to analyze data in a way that mimics the human brain, learning the intrinsic patterns and representational hierarchy of sample data so that they can recognize complex patterns from large amounts of data and generate accurate insights and predictions.

In the field of deep learning, there has been continuous exploration of new network structures in recent years, among which Kolmogorov-Arnold Networks (KANs) stand out as an intriguing innovation [1]. Based on the Kolmogorov-Arnold theorem, KANs differ from traditional Multi-Layer Perceptrons (MLPs) by replacing fixed linear weight matrices with learnable spline functions, which makes it capable of maintaining good expressive power while requiring a smaller number of parameters [2].

Convolutional Neural Networks (CNNs) proposed by LeCun et al. have been widely used in the realm of computer vision and have greatly advanced the field due to their ability to process data with high dimensionality such as images and videos [3,4]. These networks commonly use linear transformations to process image data and may incorporate activation functions in the convolutional layer to help the network discern spatial relationships within data, which considerably reduces the total number of parameters required to recognize intricate patterns.

While CNNs is effective, one notable bottleneck is their reliance on fixed activation functions, which may restrict the network's ability to capture complex patterns adaptively. Convolutional Kolmogorov–Arnold Networks (CKANs) proposed by Alexander allows CNNs gain advantages from adaptability and less complex parameterization offered by KANs [5]. Incorporating spline-based convolutional layers, as introduced by M. Fey and J. E. Lenssen, enhances the network's capacity to effectively model non-linear relationships [6].

The better interpretability and expressiveness of KANs, and CKANs usually means that it is more difficult to optimize them [7]. Since both models fit the objective function by updating the parameters in the spline function, the choice of the spline function is critical. However, the selection and interpretation of the spline function in different cases is still being investigated. Although KANs and CKANs have great expressive potential compared to traditional network structures, they do not significantly outperform MLPs and CNNs currently [8]. This paper combines KANs and MLPs, as well as CKANs and CNNs, respectively, to investigate different network structures in detail and compare their performances with the traditional models. This work aims to find networks that require fewer parameters while achieving an accuracy comparable to existing conventional network structures.

2. Method

2.1. Theoretical foundation of Kolmogorov-Arnold Networks (KANs)

The essence of KANs lies in their distinctive architectural design which is based on the Kolmogorov-Arnold representation theorem [1,9]. In contrast to the conventional MLPs which employ fixed activation functions at each node, KANs boast an innovative approach by situating learnable activation functions along network boundaries. This pivotal transition from static to dynamic node functionalities is achieved by eschewing the traditional linear weight matrices in favor of malleable spline functions. These splines are meticulously parameterized and fine-tuned throughout the training process, thereby endowing the model with an enhanced flexibility and adaptability to intricate data patterns.

2.2. Theoretical foundation of convolutional Kolmogorov-Arnold networks (CKANs)

Convolutional Kolmogorov-Arnold Networks (CKANs) share similarities with their CNN counterparts but distinguish themselves primarily in the construction of their convolutional kernels [4]. While CNNs utilize a kernel composed of fixed weights, CKANs employ a kernel where each element is a trainable nonlinear function leveraging B-Splines. In this paradigm, traditional convolutional layers are supplanted by KAN convolutional layers, which, post flattening, may be succeeded by either a KAN or an MLP. A defining advantage of Convolutional KANs is their reduced parameter count relative to other architectures. This efficiency stems from the inherent capacity of B-Splines to smoothly approximate complex activation functions that conventional Rectified Linear Unit (ReLU) activations cannot match within the convolutional framework.

2.3. Evaluation metrics

The efficacy of the proposed Networks is quantified through a suite of established evaluation criteria that furnish a multifaceted review of the model's capabilities. The selected metrics are standard in the field of machine learning and are detailed below:

Test Accuracy: This metric reflects the proportion of total predictions that are correct on the test dataset, serving as a direct measure of the model's classification accuracy.

Test Precision: Precision is the ratio of true positive predictions to the total number of positive predictions made, offering insight into the model's ability to avoid false positives.

Test Recall: Also known as sensitivity, recall measures the ratio of true positive predictions to all actual positive instances. It is a critical indicator of the model's capability to detect the positive class.

DOI: 10.54254/2755-2721/83/2024GLG0060

Test F1 Score: The F1 Score is the harmonic mean of precision and recall, providing a single score that balances the trade-off between these two metrics. It is particularly informative for datasets with class imbalance.

Number of Parameters: The complexity of the model is gauged by the total count of trainable parameters. This metric is indicative of the model's capacity for learning and its potential susceptibility to overfitting.

3. Experiments and results

3.1. Experimental settings

In this section, the experimental settings are demonstrated and the results are evaluated to assess the performance of models with KAN Convolutional Layers and KAN layers in comparison with standard convolutional neural networks and MLP networks, as well as comparing KAN layers to traditional fully-connected layers. For the experiments, the CIFAR-10 dataset is selected as the basis for evaluation [10]. This work trained various architectures on this dataset to compare their performance. To achieve this, this paper proposed models that integrate both Linear and Kan Linear layers, as well as classical Convolutional and Kan Convolutional layers. Figure 1 and Figure 2 below presents the diverse architectures of KAN and KAN Convolutional Layers that were utilized in the experiments, respectively.

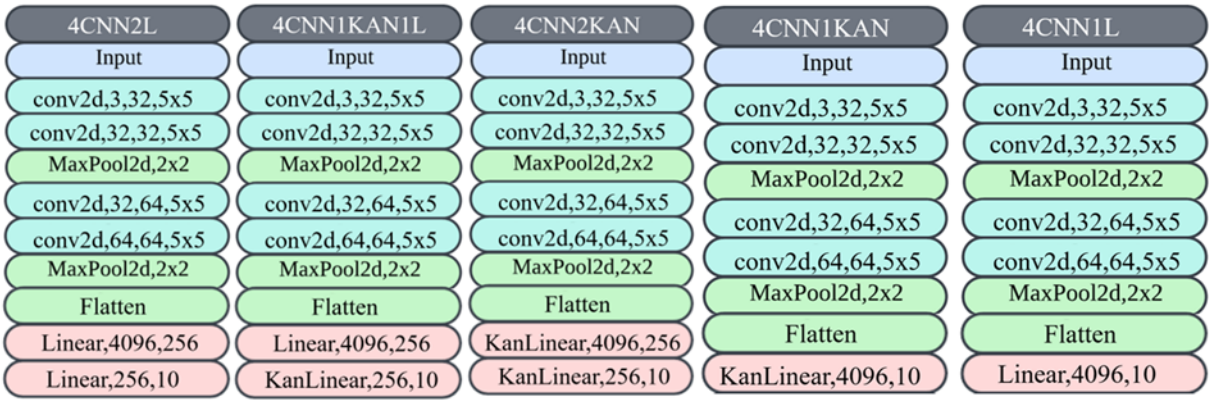

Figure 1. Architectures of KAN-based models (Figure Credits: Original).

The first branch of architectures used in experiments, focusing on the design of fully connected layers. This design keeps the convolutional layer part of the different architectures the same, which are all four-layer CNNs, and make adjustments to the fully connected layer. Here the author considers networks with one and two fully connected layers, respectively, as demonstrated in Figure 1. For architectures with only one fully connected layer, this work designs networks with one MLP layer (4CNN1L) and one KAN layer (4CNN1KAN), respectively. For architectures with two fully connected layers, this work designs networks with two layers of MLP (4CNN2L), two layers of KAN (4CNN2KAN), and one layer of KAN and one layer of MLP (4CNN1KAN1L), respectively. In the case of fully connected structures with one and two layers, respectively, this work designs and compares different architectures to determine how to combine the KAN structure and the classical MLP structure to achieve optimal performance.

The second branch of network architectures used in the experiments, focusing on the structural design of convolutional layers, as displayed in Figure 2. This work keeps the fully-connected layers of the different architectures the same, both two-layer MLPs, and make adjustments and modifications to the convolutional layers. Here this work considers networks with three convolutional layers. This work designs networks with three CNN layers (3CNN2L), one CKAN and two CNN layers (1CKAN2CNN2L), and two CKAN and one CNN layer (2CKAN1CNN2L), respectively. This work

DOI: 10.54254/2755-2721/83/2024GLG0060

designs and compares different architectures to determine how to combine the KAN convolutional structure and the classical CNN structure to achieve optimal performance.

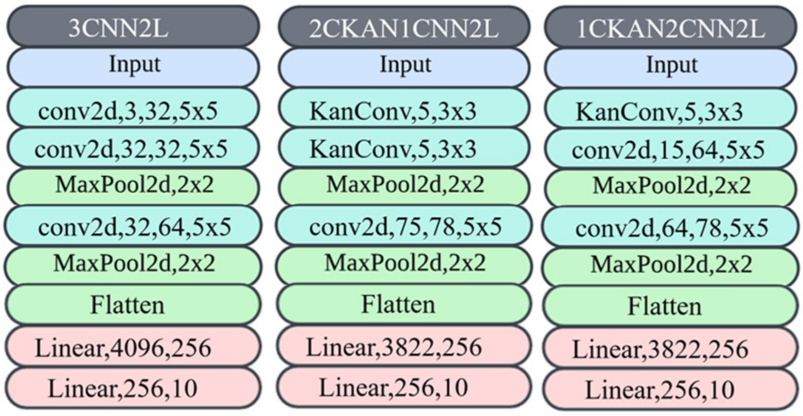

Figure 2. Architectures of CKAN-based models (Figure Credits: Original).

3.2. Quantitative comparison between KAN and MLP

Table 1 presents an evaluation of performance criteria including accuracy, precision, recall, F1 score and parameter count for a range of models with combinations of KANs and/or MLPs tested on the CIFAR-10 Dataset. This evaluation offers insights into proficiency and potency of distinct model setups.

Table 1. Performance comparison between KAN-based and MLP-based models on CIFAR-10.

Model	Accuracy	Precision	Recall	F1 Score	#Params
4CNN2KAN	76.98%	77.09%	76.98%	76.98%	15850K
4CNN1KAN1L	74.59%	74.65%	74.59%	74.54%	15814K
4CNN2L	75.00%	75.07%	75.00%	74.93%	1134K
4CNN1KAN	76.39%	76.47%%	76.39%	76.31%	697K
4CNN1L	74.74%	74.79%	74.74%	74.68%	124K

As observed in Table 1, in the CIFAR-10 dataset, the network with a single KAN layer(1KAN) not only surpassed the two-layer MLP(2L) structure in accuracy, reaching 76.98%, but also exhibited a more parameter-efficient design, holding 697K parameters compared to the 1134K of the two-layer MLP.

In comparison to the network with only one MLP layer (1L), the configuration that includes a KAN layer shows a roughly 2% gain in accuracy, even though it has a higher parameter count.

Besides,compared to the network with two layers of MLP, the fully connected networks integrated with KAN, namely 1KAN1L and 2KAN, have achieved better performance with accuracy rates of 74.59% and 76.98% respectively. However, they require a greater number of parameters.

3.3. Quantitative comparison between CKAN and CNN

As the Table 1 presented in the dimension of KAN and MLP combinations, the Table 2 presents an evaluation of performance criteria including accuracy, precision, recall, F1 score and parameter count for the proposed models with combinations of KAN Convolutions and/or classical convolutions tested on the CIFAR-10 Dataset.

Table 2. Performance comparison between CKAN-based and CNN-based models on CIFAR-10.

Model	Accuracy	Precision	Recall	F1 Score	#Params
3CNN2L	66.36%	66.42%	66.36%	66.31%	1130K
2CKAN1CNN2L	69.07%	69.07%	69.07%	69.84%	1128K
1CKAN2CNN2L	70.00%	71.02%	70.00%	70.30%	1130K

As observed in Table 2, in the CIFAR-10 dataset, compared to the network with three layers of CNNs, the network with two layers of KAN and one layer of CNN achieves a greater advantage with 69.07% accuracy while maintaining the same number of parameters. Also, the network with one layer of KAN and two layers of CNN exceeds with an accuracy of 70% with the same number of parameters.

4. Discussion

In comparing KAN and MLP through the combination of different network architectures, a single-layer KAN can achieve better accuracy while maintaining a smaller number of parameters compared to a two-layer MLP. This confirms the theory that introducing learnable spline functions can make fully connected networks more interpretable [2]. Although KAN itself requires more parameters than MLP when maintaining the same number of layers due to its inherent characteristics, the current experimental results have demonstrated that it is feasible to reduce the number of parameters by designing networks with fewer KAN layers than those needed to achieve the same performance with an MLP.

Similarly, introducing KAN convolutional layers in the convolutional structure can improve the performance of the model without increasing the number of parameters.

Overall, by designing experiments to compare different networks with KANs and KAN convolutions, it could be found that the structures with KAN and KAN convolutions were able to achieve higher accuracy while maintaining the number of parameters, or maintain accuracy with a lower number of parameters. In fully connected networks, a KAN structure with only one layer can achieve higher accuracy with fewer parameters than a traditional two-layer MLP structure. In convolutional structures, the research found that replacing one or two layers of classical convolutional layers with KAN convolutional layers can achieve higher accuracy while maintaining the number of parameters.

5. Conclusion

This research has demonstrated that integrating KANs and CKANs into traditional neural network architectures can lead to significant improvements in accuracy and parameter efficiency. The comparative analysis on the CIFAR-10 dataset revealed that a single KAN layer outperforms a two-layer MLP in terms of accuracy while requiring fewer parameters. Similarly, the incorporation of KAN convolutional layers into CNNs showed comparable or superior performance without an increase in parameter count. These findings suggest that KAN-based models offer a promising avenue for developing more efficient and interpretable deep learning models.

The results highlight the need for further exploration into the optimization and application of KANs and CKANs across various domains. Subsequent research should concentrate on enhancing the scalability and robustness of these architectures, ensuring their potential is fully realized in complex real-world scenarios. With continued innovation, KANs and CKANs could become pivotal in advancing the field of deep learning.

References

[1] Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., et, al. (2024). Kan: Kolmogorov-arnold networks. arXiv preprint arXiv:2404.19756.
[2] Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. Neural networks, 2(5), 359-366.
[3] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.
[4] Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. IEEE transactions on neural networks and learning systems, 33(12), 6999-7019.
[5] Bodner, A. D., Tepsich, A. S., Spolski, J. N., & Pourteau, S. (2024). Convolutional Kolmogorov-Arnold Networks. arXiv preprint arXiv:2406.13155.

DOI: 10.54254/2755-2721/83/2024GLG0060

[6] Fey, M., Lenssen, J. E., Weichert, F., & Müller, H. (2018). Splinecnn: Fast geometric deep learning with continuous b-spline kernels. In Proceedings of the IEEE conference on computer vision and pattern recognition, 869-877.
[7] Drokin, I. (2024). Kolmogorov-Arnold Convolutions: Design Principles and Empirical Studies. arXiv preprint arXiv:2407.01092.
[8] Jamali, A., Roy, S. K., Hong, D., Lu, B., & Ghamisi, P. (2024). How to Learn More? Exploring Kolmogorov-Arnold Networks for Hyperspectral Image Classification. arXiv preprint arXiv:2406.15719.
[9] Arnold, V. I. (2009). On functions of three variables. Collected Works: Representations of Functions, Celestial Mechanics and KAM Theory, 1957–1965, 5-8.
[10] CIFAR-10 Dataset. URL: https://www.cs.toronto.edu/~kriz/cifar.html. Last Accessed: 2024/08/24