J. Electrical Systems 20-10s (2024): 6274-6295

[1] Mustafa Umut DEMiREZEN

[2]Ekin Can ERKUŞ

Kolmogorov-Arnold Networks for Accurate Forecasting of User Actions in Complex Digital Ecosystems

Journal of Electrical Systems

Abstract: - In this paper, we propose a novel hybrid neural network architecture, the KAN-enhanced QResNet, specifically designed to improve user behavior prediction in complex digital ecosystems such as e-commerce platforms. The proposed model integrates the feature decomposition capabilities of Kolmogorov-Arnold Networks (KAN) with the quadratic residual interaction modeling of Quadratic Residual Networks (QResNet). This combination allows the model to capture both compositional structures and higher-order feature interactions, leading to enhanced predictive performance. We rigorously evaluated the KAN-enhanced QResNet on three diverse e-commerce datasets: E-Commerce Customer Churn Data, Multi-Category Store Behavior, and Customer Churn Analysis, demonstrating its superiority over traditional machine learning models such as Logistic Regression, Support Vector Machines (SVM), XGBoost, and standard Feedforward Neural Networks (FF-MLP). The KAN-enhanced QResNet achieved a significant F1-score of 0.86 on the E-Commerce Customer Churn Data, outperforming Logistic Regression (F1-score: 0.68) and XGBoost (F1-score: 0.77). Similarly, the model attained an F1-score of 0.79 on the Multi-Category Store Behavior dataset, compared to 0.69 achieved by XGBoost. These results demonstrate the model's superior ability to capture complex non-linear interactions, a challenge for conventional models. Our contributions include: (1) Introducing a novel hybrid architecture that integrates KAN and QResNet to efficiently handle high-dimensional data and capture intricate feature interactions; (2) Comprehensive empirical validation across three real-world datasets, showcasing the robustness and adaptability of the model; (3) Establishing the KAN-enhanced QResNet as a state-of-the-art solution for user behavior prediction tasks, outperforming traditional models in terms of accuracy, precision, recall, and F1-score. The findings of this study highlight the transformative potential of hybrid neural network architectures for predictive modeling in complex digital environments and offer new directions for future research in domains such as customer churn prediction, personalized recommendation systems, and real-time marketing strategies.

Keywords: Neural Network, Kolmogorov-Arnold Networks, Multilayer Perceptron, Quadratic Residual Neural Networks, User Behavior, KAN, MLP.

I. INTRODUCTION

In the rapidly evolving landscape of digital ecosystems, understanding and predicting user behavior has become a critical challenge for businesses, marketers, and platform developers alike. The ability to accurately forecast user actions can significantly enhance user experience, optimize content delivery, and drive engagement across various digital platforms. However, the complexity and non-linear nature of user interactions in these environments have often eluded traditional predictive models, necessitating more sophisticated approaches.

In modern digital ecosystems, user behavior prediction is crucial for improving user experiences, optimizing content delivery, and maximizing engagement across platforms such as social media, e-commerce, and online learning environments. These ecosystems produce vast amounts of data from user interactions, creating opportunities and challenges for businesses that seek to predict user actions such as clicks, purchases, or content engagement. However, the complexity of these digital environments, combined with non-linear and multidimensional user interactions, often limits the effectiveness of traditional machine learning models.

Machine learning models like Support Vector Machines (SVMs), logistic regression, XGBoost, Multilayer Perceptrons (MLPs), and Random Forests have been widely used to predict user behavior. However, they often struggle with modeling nonlinear relationships and high-dimensional data. As user behavior becomes increasingly dynamic and influenced by multiple factors, advanced methods like Kolmogorov-Arnold Networks (KANs) and Quadratic Residual Networks (QResNets) have emerged as promising alternatives to traditional models. These newer architectures offer the ability to model higher-order interactions, capture non-linearity, and handle the complexities of large-scale, dynamic digital ecosystems.

Support Vector Machines (SVMs) have been applied in a variety of user behavior prediction tasks, particularly in domains like e-commerce and customer churn prediction. While SVMs are highly effective in separating binary

[1] Intelligent Application DC, Huawei Turkey R&D Center, Huawei Türkiye R&D Center

[2] Intelligent Application DC, Huawei Turkey R&D Center, Huawei Türkiye R&D Center

classes by maximizing the margin between data points [1], they have limitations when applied to high-dimensional and non-linear data. Studies have shown that SVMs often underperform in environments where user behavior involves complex interactions and temporal dependencies [2]. Additionally, SVMs are computationally expensive, particularly when working with large datasets, which limits their scalability in real-time applications [3].

Logistic regression, while still widely used for predicting binary outcomes like click-through rates (CTR) or purchase likelihood, is inherently linear and lacks the flexibility needed to model non-linear user behavior interactions. It works well as a baseline model, but recent studies have found that logistic regression often falls short in terms of accuracy when applied to more complex user behavior prediction tasks [4]. Furthermore, logistic regression requires careful feature engineering to capture complex relationships between user actions and features, making it inefficient in large-scale, unstructured datasets.

XGBoost, a widely used gradient boosting algorithm, has demonstrated high performance in various user behavior prediction tasks, including e-commerce personalization and customer segmentation [5]. XGBoost excels in capturing feature interactions by building decision trees iteratively, but it still struggles with modeling temporal dependencies and evolving user preferences over time [6]. Its reliance on predefined tree structures makes it less suitable for tasks involving dynamic user behavior, which requires more flexible architectures that can adapt to rapidly changing data patterns [7].

Multilayer Perceptrons (MLPs) have been widely adopted in user behavior prediction due to their universal function approximation capabilities. MLPs can capture non-linear relationships between input features, making them more flexible than linear models like logistic regression [8]. However, MLPs are prone to overfitting, especially when trained on small or imbalanced datasets [9]. Furthermore, they require large amounts of data to generalize well, which limits their practical application in real-time user behavior prediction tasks.

Random Forests have also been extensively used for customer behavior prediction tasks, such as predicting customer churn and user segmentation [10]. Random Forests offer high interpretability and are less prone to overfitting due to their ensemble structure. However, like XGBoost, they are not well-suited to capturing temporal dependencies or handling non-linear, evolving user behavior. Random Forests also require significant computational resources, particularly when applied to large datasets, which makes them less practical for real-time decision-making [11].

While traditional machine learning models have proven effective in certain tasks, they face significant limitations when applied to the complexities of user behavior prediction in dynamic digital ecosystems. First, these models often fail to capture nonlinear interactions between features, which are crucial for accurately predicting user actions. Models like logistic regression, SVMs, and decision trees typically assume linear relationships, requiring extensive feature engineering to handle non-linearities. Second, the high dimensionality of user data presents a challenge for traditional models, as they struggle to handle the large number of features and interactions present in user behavior data. Third, existing models are prone to overfitting when trained on sparse or imbalanced datasets, especially in dynamic environments where user preferences change over time. Finally, traditional models lack the adaptability needed to handle real-time predictions in rapidly evolving ecosystems, limiting their effectiveness in tasks such as dynamic content recommendation or targeted advertising [12].

Kolmogorov-Arnold Networks (KANs) offer a robust solution to these challenges by leveraging the Kolmogorov-Arnold representation theorem, which states that any continuous multivariate function can be represented as a composition of univariate functions [13]. This allows KANs to decompose complex multivariate functions into simpler components, making them well suited to capturing the intricate, non-linear interactions present in user behavior data. Unlike traditional models, KANs can handle high-dimensional data more effectively by breaking down the data into lower-dimensional subcomponents, allowing for better generalization and more accurate predictions [13]. Moreover, KANs have been shown to outperform traditional models in tasks where capturing high-order feature interactions is crucial, such as CTR prediction and content recommendation [7].

Despite their advantages, KANs are not without limitations. The computational complexity of KANs can be high, particularly when applied to large datasets with many features. Additionally, KANs are prone to overfitting when not properly regularized, especially in cases where the training data is sparse or imbalanced. These limitations highlight the need for further research into optimization techniques that can make KANs more efficient and scalable for practical applications in digital ecosystems [1].

Quadratic Residual Networks (QResNets) introduce an additional layer of nonlinearity to the traditional deep learning architecture by incorporating quadratic residuals into the network [14]. This allows QResNets to model more complex interactions between features without increasing the depth of the network, making them more efficient than traditional deep learning models like MLPs [14]. QResNets have been particularly effective in solving high-frequency prediction tasks, such as Physics-Informed Neural Networks (PINNs), where traditional models often struggle. The added quadratic terms in QResNets allow the model to capture higher-order interactions between variables, which can significantly improve predictive accuracy in tasks involving complex user behavior data.

However, the inclusion of quadratic terms also increases the risk of overfitting, particularly when training on small or noisy datasets. QResNets also require more computational resources than simpler architectures, as the quadratic terms add complexity to the model. This can make them less practical for real-time applications where computational efficiency is critical. Despite these limitations, QResNets provide a significant advantage in terms of expressiveness and predictive power, particularly in high-dimensional data settings [14].

This study addresses the limitations of traditional machine learning models in accurately predicting user behavior in complex digital ecosystems. Specifically, we aim to solve the problem of capturing nonlinear, high-dimensional interactions and temporal dependencies in user data, which are often missed by traditional models like SVMs, logistic regression, and random forests. The dynamic nature of digital ecosystems, where user preferences and interactions evolve rapidly, requires a more flexible and adaptive approach to predictive modeling.

To address these challenges, we propose a novel neural network architecture that integrates Kolmogorov-Arnold Networks (KANs) with Quadratic Residual Networks (QResNets). Our hypothesis is that this hybrid architecture will outperform traditional models by capturing higher-order feature interactions and providing more accurate and efficient predictions of user actions. Specifically, we hypothesize that:

- Hypothesis 1: The integration of KANs and QResNets will lead to significant improvements in predictive accuracy compared to traditional machine learning models, particularly in tasks involving high-dimensional and non-linear user behavior data.
- Hypothesis 2: The hybrid KAN-QResNet architecture will provide better generalization performance, reducing the risk of overfitting when trained on sparse or imbalanced datasets.
- Hypothesis 3: The proposed model will demonstrate superior computational efficiency in handling large-scale, dynamic datasets, making it suitable for real-time applications in digital ecosystems.

The methodology for this study involves the development of a hybrid KAN-QResNet model that combines the strengths of Kolmogorov-Arnold Networks in handling high-dimensional data with the ability of Quadratic Residual Networks to capture higher-order feature interactions. We will train this model on large-scale datasets from various digital ecosystems, including ecommerce platforms, social media networks, and online learning environments. The datasets will be preprocessed to normalized data, handle missing values, and engineer relevant features. The training process will involve extensive hyperparameter tuning and regularization techniques to ensure that the model generalizes well to unseen data. The performance of the hybrid KAN-Enhanced QResNet model will be evaluated using standard metrics, such as accuracy, precision, recall, and F1-score.

The hybrid KAN-QResNet model is rigorously evaluated using three comprehensive real-world datasets drawn from diverse digital ecosystems: E-commerce Customer Churn Analysis and Prediction [15], E-commerce Behavior Data from Multicategory Store [16], and E-commerce Customer Churn [17]. These datasets offer detailed and diverse user behavior patterns, encompassing key factors such as purchasing habits, content consumption, and social interactions. The experimental design integrates a robust comparative analysis between the KAN-QResNet architecture and traditional machine learning models, including SVMs, logistic regression, XGBoost, and MLPs. The goal is to demonstrate the hybrid model's significant improvements in capturing complex, nonlinear relationships in user data, yielding higher predictive accuracy, superior generalization, and enhanced computational efficiency.

Combining KAN's strength in modeling high dimensional data through decomposition and QResNets ability to capture higher order feature interactions, the KAN-QResNet architecture provides a strong framework to address the complexities of modern user behavior data. These datasets are not only abundant in terms of user interactions and features but also provide a near-real-world setup enabling us to measure the model's efficacy in real-life, high impact applications such as customer churn prediction, product recommendation, dynamic content delivery etc. With the help of this comparative analysis, it also uses for both how well the model does in real-time environments and benefits of its computational performance over conventional models. This examination highlights the model's ability to reshape the fundamental standards of predictive modeling throughout digital ecosystems, by being able provide a more light and fast response method for related with complex behavior prediction.

As far as we know, this is the first work to design a new neural network architecture that integrates Kolmogorov-Arnold Network (KANs) with Quadratic Residual Networks (QResNets) for capturing higher-order interactions in user behavior prediction. The combination of these two strong models helps us to consider the complicated and non-linear relationships we see in user behavior data that are usually ignored by traditional models. The possible hybrid architecture is a step forward in the realm of prediction modeling digital ecology and it forms powerful solutions for real-time applications which include dynamic content recommendation, targeted advertising etc.

The rest of the paper is organized as follows: Section II discusses related work on predicting user behavior using KANs and QResNets. In Section III, we explain the methodology that includes the architecture for the hybrid KAN-QResNet model in detail. In Section IV, we present details of the experimental setup (datasets, training procedure and evaluation metrics). The results, comparing to the performance of hybrid model with traditional machine learning models are presented in Section V. In Section VI, we present and discuss these results, while in Section VII, we draw our conclusions about the contributions of this research and some possible directions for future lines.

II. Background And Related Work

2.1 User Behavior Prediction in Digital Ecosystems

Predicting user behavior in digital ecosystems has been a significant area of research in recent years. Traditional approaches have often relied on statistical methods and machine learning algorithms to forecast user actions based on historical data. For instance, Liang et al. [18] employed collaborative filtering techniques to predict user preferences in e-commerce settings, while Zhang et al. [3] utilized deep learning models to forecast user engagement on social media platforms. Traditional approaches often involve the use of decision trees, support vector machines (SVMs), and logistic regression models, which are designed to identify patterns in user interaction data and predict future actions. For instance, Baeza-Yates [19] discusses the application of decision trees and logistic regression in predicting user clicks and engagement on web pages, highlighting the limitations of these models in capturing complex, non-linear relationships in user behavior data.

Neural networks have also been extensively applied in this domain. In particular, fully connected networks and convolutional neural networks (CNNs) have been used for tasks such as click-through rate prediction and recommendation systems. However, these models often require extensive tuning and large datasets to perform effectively, and they may struggle with overfitting, especially in high-dimensional spaces [20].

2.2 Advances in Non-linear Modeling Techniques

As the limitations of traditional machine learning models have become apparent, research has increasingly focused on more advanced non-linear modeling techniques. Deep learning models, including recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), have shown significant promise in capturing temporal dependencies and sequential patterns in user data. These models have been particularly effective in session-based recommendation systems and predictive analytics, where understanding the sequence of user actions is critical [21], [22].

However, deep learning models present their own set of challenges, such as the need for large datasets, high computational costs, and difficulties in interpretability. The complexities of these models make them less suitable for scenarios where model transparency and efficiency are paramount [23], [24].

2.3 The Kolmogorov-Arnold Representation Theorem

The Kolmogorov-Arnold representation theorem, also known as the superposition theorem, is a fundamental result in the theory of function approximation. Kolmogorov [25] initially proved in 1957 that any continuous function of several variables could be represented as a superposition of continuous functions of one variable and addition. Arnold [26] later refined this theorem, providing a more constructive proof.

The theorem states that for any continuous function $f: [0,1]^n \to \mathbf{R}$, there exist continuous functions ψ and ϕ_{ij} such that:

$$f(x_1, \dots, x_n) = \sum_{i=1}^{2n+1} \psi\left(\sum_{j=1}^{n} \phi_{ij}(x_j)\right) \tag{1}$$

This powerful result suggests that complex multivariate functions can be decomposed into simpler, univariate components, which has significant implications for function approximation and machine learning. While the Kolmogorov-Arnold theorem is primarily a theoretical result, researchers have explored its potential applications in machine learning and function approximation. Kurkova [27] discussed the relationships between Kolmogorov's theorem and neural network architectures, highlighting the theoretical foundations for universal approximation capabilities of certain network structures.

Poggio and Girosi [28] proposed regularization networks, which drew inspiration from Kolmogorov's theorem to develop a framework for approximating multivariate functions. Their work laid the groundwork for exploring how the principles of the theorem could be applied in practical machine learning contexts. Despite the theoretical promise, directly implementing Kolmogorov-Arnold Networks (KANs) has proven challenging. The functions ψ and ϕ_{ij} in the theorem are generally not smooth or easily computable, making practical implementations difficult. Braun and Griebel [29] discussed these challenges and proposed alternative approaches to leveraging the theorem's principles in computational settings.

2.4 Recent Advances in Complex Function Approximation

Recent research has focused on developing novel neural network architectures that can effectively model complex, nonlinear relationships in high-dimensional data. For example, Montúfar et al. [30] explored the expressive power of deep neural networks, showing how certain architectures can approximate a wide range of functions efficiently.

In the context of user behavior prediction, Ma et al. [31] proposed a hierarchical attention network for modeling sequential user behaviors in e-commerce environments, demonstrating improved performance over traditional methods.

The present study builds upon these foundations, exploring how the principles inspired by the Kolmogorov-Arnold theorem can be applied to create more effective models for user behavior prediction in diverse digital ecosystems.

2.5 Implications for Personalized Content Delivery and User Experience

The superior predictive performance of Kolmogorov-Arnold Networks (KANs), as demonstrated by their ability to accurately forecast user actions such as clicks, purchases, and content engagement, has profound implications for personalized content delivery and the enhancement of user experience across digital platforms. By leveraging large datasets of user activity, KANs identify intricate patterns in user behavior that traditional machine learning models often fail to detect, enabling more precise predictions of individual user preferences and behaviors.

Personalized Content Delivery: One of the primary applications of KANs in digital ecosystems is the optimization of content delivery. By understanding user preferences at a granular level, KANs allow platforms to tailor content recommendations more effectively. For example, in streaming services like Netflix or music platforms like Spotify, KANs can analyze historical user data to predict what movies or songs a user is most likely to enjoy next, leading to a more engaging and satisfying user experience [32], [33].

Targeted Advertising: KANs also hold significant potential for improving the effectiveness of targeted advertising. By accurately predicting which ads are most likely to resonate with a specific user, based on their past interactions and inferred preferences, advertisers can deliver more relevant ads, reducing wastage and increasing return on investment (ROI). This capability is especially valuable in environments where user attention is a scarce resource, such as social media platforms like Facebook or Instagram, where personalized ads can lead to higher engagement rates and conversions [34], [35].

Another critical application of KANs is in the optimization of user interfaces (UI). By predicting how users interact with different elements of a website or app, KANs can inform the design of more intuitive and user-friendly interfaces. For instance, e-commerce platforms can use KANs to anticipate which layout or navigation options will lead to higher conversion rates, thereby enhancing the overall shopping experience [36], [37].

The ability of KANs to provide personalized content and targeted advertising translates directly into a richer and more personalized user experience. Users are more likely to engage with content that feels tailored to their interests and needs, leading to increased satisfaction and loyalty. Furthermore, as KANs continually learn from user behavior, they enable platforms to adapt to changing user preferences over time, ensuring that the user experience remains relevant and engaging [38], [20].

In summary, the deployment of KANs in digital ecosystems not only improves the precision of user behavior predictions but also drives significant improvements in personalized content delivery, targeted advertising, and UI optimization. These advancements contribute to a more engaging, satisfying, and ultimately profitable user experience, making KANs an invaluable asset in the management of digital platforms.

2.6 Problem Statement

Accurately forecasting user actions in complex digital ecosystems presents significant challenges due to the high-dimensionality, non-linearity, and dynamic nature of user behavior. Existing machine learning approaches, including traditional models such as Support Vector Machines (SVMs), logistic regression, and gradient boosting machines, as well as more recent deep learning architectures, struggle to effectively capture the intricate interactions between various behavioral factors. This is largely due to their inherent limitations in modeling high-order, non-linear relationships within large-scale datasets.

At the core of this problem lies the difficulty in modeling user interactions, which often involve multiple variables interacting in a complex, non-linear fashion. These interactions evolve over time, leading to temporal dependencies that further complicate the predictive task. Traditional approaches assume that user actions can be modeled as linear or low-order combinations of individual features. However, such assumptions fail to capture the multi-dimensional and high-order feature interactions present in real-world data.

Mathematically, let $\mathbf{X} \in \mathbb{R}^{n\times d}$ represent a dataset of n user interactions, where each interaction consists of d features. The goal is to predict user actions $\mathbf{y} \in \mathbb{R}^{n}$, which are binary or continuous variables representing user decisions, such as clicks, purchases, or engagements. Standard machine learning models attempt to learn a mapping $f: \mathbb{R}^{d} \to \mathbb{R}$ such that:

$$\mathbf{y} = f(\mathbf{X}) + \epsilon \quad (2)$$

where ϵ denotes noise in the data. However, these models typically assume that f is a low-order polynomial or linear function, which severely limits their ability to capture complex interactions between features.

Kolmogorov-Arnold Networks (KANs) provide a theoretical solution to this problem by leveraging the Kolmogorov-Arnold representation theorem, which asserts that any continuous multivariate function can be decomposed into a finite sum of univariate functions:

$$f(\mathbf{X}) = \sum_{i=1}^{d} \phi_i\left(\sum_{j=1}^{d} \psi_{ij}(x_j)\right) \quad (3)$$

where ϕ_i and ψ_{ij} are univariate continuous functions. This decomposition allows KANs to model the high-dimensional, nonlinear relationships between features without the limitations of traditional approaches.

Quadratic Residual Networks (QResNets) further enhance the expressive power of deep learning models by introducing quadratic residuals, enabling the network to capture higher-order interactions within each layer. The quadratic term is represented as:

$$\mathbf{y} = \sigma(W_2\mathbf{x} \circ W_1\mathbf{x} + W_1\mathbf{x} + b) \quad (4)$$

where $\circ$ denotes the Hadamard product, and W_1, W_2 are trainable weight matrices. By incorporating quadratic terms, QResNets are able to capture interactions of the form $x_i x_j$, which are essential for modeling complex dependencies between features.

Despite these advancements, challenges remain in efficiently integrating KANs and QResNets into a unified framework that can handle the scale and complexity of real-world digital ecosystems. The primary research question addressed by this paper is: How can the combination of KANs and QResNets improve the predictive accuracy and efficiency of user action forecasting in complex digital environments?

This work seeks to address these challenges by developing a hybrid architecture that combines the decomposition power of KANs with the high-order interaction modeling of QResNets, providing a more robust solution to the problem of user behavior prediction in dynamic, high-dimensional data environments.

III. Theoretical Framework for Methodology

User behavior in digital ecosystems is inherently complex and multidimensional, involving numerous interactions across various platforms, devices, and contexts. Traditional machine learning models, such as logistic regression, support vector machines, and even conventional neural networks, often struggle with the intricacies of user behavior data, particularly in terms of capturing the non-linear and temporal dependencies that characterize user interactions.

KANs, by contrast, are well-suited to address these challenges. The architecture of KANs allows them to capture intricate patterns in user behavior data by modeling the interactions between multiple variables more effectively. This is particularly beneficial when dealing with high-dimensional data, such as clickstream data, purchase histories, or social media interactions, where traditional models might either oversimplify or entirely miss critical patterns [8].

3.1 Theoretical Foundation of Kolmogorov-Arnold Networks

At a fundamental level, Kolmogorov-Arnold Networks (KANs) are based on the Kolmogorov-Arnold superposition theorem stating that any multivariate continuous function can be expressed as the sum of univariate functions. That theorem grounds a set of highly expressive neural network architectures in the language of well-defined, clear mathematical terms. KANs approximate any continuous input to output curve by decomposing the joint multivariate relationship between inputs into modes consisting of univariate functions, which enable delivering approximations that capture a combination of non-linearities, so traditional machine learning models miss.

The architecture of KANs is designed to decompose a given multivariate function $\left(f(x_1, x_2, \ldots, x_n)\right)$ into a series of simpler functions, typically of the form:

$$f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{m} g_i\left(\sum_{j=1}^{n} \phi_{ij}(x_j)\right) \quad (5)$$

Where (g_i) and (ϕ_{ij}) are univariate functions. This decomposition allows KANs to model complex interactions between variables more effectively than traditional models, which often rely on explicit feature engineering or hierarchical structures to capture such interactions [25].

3.2 Quadratic Residual Networks (QResNets)

Quadratic Residual Networks (QResNets) offer a significant advancement over traditional deep neural networks by introducing quadratic residuals at each layer to enhance the nonlinearity and expressive power of the model. Unlike plain deep neural networks (DNNs), where the nonlinearity stems solely from activation functions applied to a linear weighted sum of inputs, QResNets incorporate an additional quadratic residual term that amplifies the network's capacity to capture complex, high dimensional patterns [14]. A standard DNN layer can be expressed as:

$$y_{\text{DNN}} = \sigma(Wx + b) \quad (6)$$

where W and b are the learnable parameters, x represents the input, and σ denotes the non-linear activation function. While effective in many tasks, this architecture often requires deeper or wider networks to approximate highly complex functions accurately.

In contrast, QResNets introduce a quadratic residual term, resulting in the following formulation for a QResNet layer:

$$y_{\mathrm{QRes}} = \sigma(W_2 x \circ W_1 x + W_1 x + b) \tag{7}$$

where o denotes the Hadamard (elementwise) product, W_1 and W_2 are trainable weight matrices, and the quadratic residual term $W_2 x \circ W_1 x$ enhances the model's ability to capture intricate interactions between input features [14]. This allows QResNets to learn more complex functions with fewer parameters and shallower networks compared to traditional DNNs, as each layer has increased functional capacity.

QResNets have been shown to be more parameter-efficient than standard DNNs. The theoretical foundation underlying this efficiency stems from the fact that QResNets can approximate polynomials of higher degrees at each layer. For instance, a linearly activated QResNet of depth d can learn polynomials of degree 2^{d-1}, while a plain DNN requires substantially more layers to achieve the same level of expressiveness [14]. This capacity for efficient polynomial approximation makes QResNets particularly well-suited for tasks that require capturing higher-order interactions between variables, such as physics-informed neural networks (PINNs) and user behavior modeling in complex digital ecosystems.

To quantify the expressive power of QResNets, let $d = (d_0, d_1, \ldots, d_h)$ represent the width of each layer in the network and let σ_r be a non-linear activation function with a leading degree of non-linearity r. The functional variety of a QResNet, denoted as $V_{d,r}^{\mathrm{QRes}}$, grows exponentially with the network's depth h. In comparison, the functional variety of a standard DNN, $V_{d,r}^{\mathrm{DNN}}$, increases at a polynomial rate with respect to h, underscoring the depth and parameter efficiency of QResNets [14].

A significant benefit of QResNets is the potential to model complex, high-dimensional data with less parameters and shallower architectures. This is highly useful when a task of predicting user behavior is required, due to the fact that many times there are interactions between several inputs so as to make it worth in terms of prediction. By encoding this information in quadratic residuals, a network can be trained smaller than would be required to learn the same mapping directly form high-degree features and as such reduce the number of parameters save computation time and reduce overfitting.

Furthermore, QResNets have been shown to converge more rapidly during training compared to traditional DNNs, especially when dealing with high-frequency patterns [14]. This is due to their increased expressive power at each layer, which allows them to learn complex relationships in fewer training epochs. This rapid convergence is especially advantageous in applications requiring real-time predictions, such as e-commerce platforms, where user preferences and behaviors change dynamically. Despite their advantages, QResNets also have certain limitations. The primary drawback is the increased complexity of their computational graph, which results in longer training times per epoch compared to standard DNNs. While QResNets require fewer epochs to converge, each epoch is more computationally expensive due to the quadratic terms in each layer. This trade-off between training speed and computational cost must be carefully managed, especially in large-scale applications where computational resources may be constrained [14]. Additionally, QResNets are more prone to overfitting if not properly regularized. The added complexity of the quadratic residuals increases the model's capacity to fit noise in the data, particularly when the dataset is small or imbalanced. Techniques such as dropout, early stopping, and weight regularization are necessary to prevent overfitting in these cases.

Quadratic Residual Networks (QResNets) are a way to improve the expressiveness of neural networks with more powerful non-linearity and at the same time improving their parameter efficiency. Their power in capturing higher-order interactions has enabled such powerful methods to be applied to complex predictive tasks, one major facet of which happens in the domain of user behavior modeling in digital ecosystems. Although QResNets present some computational complexities, their fast convergence and high precision are attractive as a means for enabling further progress in the field of machine learning and artificial intelligence.

IV. Hybrid Network Architecture: Integration Of Kolmogorov-Arnold Networks And Quadratic Residual Networks

This section describes the hybrid architecture combining Kolmogorov-Arnold Networks (KAN) and Quadratic Residual Networks (QResNet). The architecture aims to model complex, non-linear, and high-order interactions between input features, which are essential for tasks like classification in complex environments. In Figure 1, the proposed hybrid architecture combines the strengths of Kolmogorov-Arnold Networks (KAN) and Quadratic Residual Networks (QResNet) to effectively model complex user behaviors in digital ecosystems. The architecture begins with an input layer that passes through the KAN layer, which is responsible for decomposing high-dimensional user features into simpler, univariate components. This decomposition allows the model to efficiently capture compositional structures inherent in the input data, such as user actions, interactions, and purchase behaviors.

After the KAN layer, we go through two branches to process a feature transformation. One connection branch leading to a QResNet layer, that quantifies the quadratic residuals in features interactions. A QResNet layer pass within it will help to capture the non-linearity in this data for the model. The other arm calculates pairwise feature interactions to help the model capture the dependencies of features. Following these transformations, a feature combination stage combines both the outputs from the QResNet layer and the pairwise interaction layer. This step ensures that the model is using both quadratic residuals and explicit feature interactions. Afterwards, the combined features are fed into an interaction aggregation layer that refines the feature interactions and aggregates them to a global representation.

The interactions are finally aggregated and fed into the classification layer where the model does its final classification task (churn = yes/no, etc). This architecture enables KAN-enhanced QResNet to work stably with highly complex, non-linear multi-dimensional user behavior data (e.g., e-commerce and other digital ecosystems). Further, detailed analysis of this novel architecture is presented in the following subsections:

4.1 KAN as Feature Extractor

Kolmogorov-Arnold Networks (KAN) function as feature extractors, transforming the input feature vector $\mathbf{x} = [x_1, x_2, \dots, x_n]^\top$ into a new feature space. This transformation is achieved by applying spline functions to each input feature. Mathematically, this transformation is represented as:

$$\mathbf{h}_{\text{KAN}} = \Phi(\mathbf{x}) = [\phi_1(x_1), \phi_2(x_2), \dots, \phi_n(x_n)]^\top \quad (8)$$

In Eq. (8), $\Phi(\mathbf{x})$ denotes the non-linear mapping applied to the input vector, and each $\phi_i(x_i)$ is a spline function applied to the i-th feature. Spline functions allow the model to capture non-linearities and higher-order patterns in the data, making them particularly suitable for tasks where the underlying relationships are non-linear.

The output of the KAN layer, $\mathbf{h}_{\text{KAN}}$, becomes a set of non-linearly transformed features that form the basis for further enhancement in the QResNet layers.

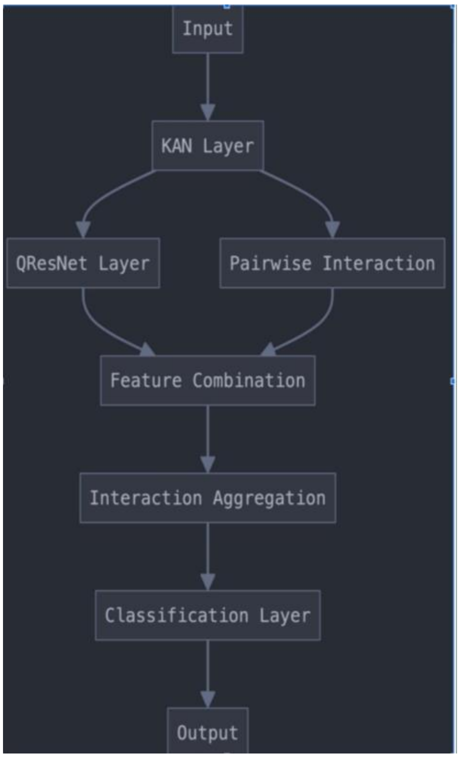

Figure 1. Proposed Architecture

4.2 QResNet as Feature Enhancer

After feature extraction by the KAN layer, the QResNet blocks are applied to enhance the non-linear interactions between the features. QResNet introduces quadratic residual connections that are specifically designed to capture high-order interactions between the spline-transformed features. The transformation of these features in QResNet is expressed as:

$$\mathbf{h}_{\text{QRes}} = \sigma(W_2\mathbf{h}_{\text{KAN}} \circ W_1\mathbf{h}_{\text{KAN}} + W_1\mathbf{h}_{\text{KAN}} + \mathbf{b}) \tag{9}$$

In Eq. (9), W_1 and W_2 are learnable weight matrices, and $\circ$ represents the Hadamard (elementwise) product, which is used to model interactions between different elements of the transformed feature vector. The activation function $\sigma(\cdot)$ (such as ReLU) introduces non-linearity, and $\mathbf{b}$ is the bias term.

This quadratic residual formulation allows QResNet to not only capture direct relationships between the features but also to model second-order interactions between features, which are crucial for understanding complex dependencies. The Hadamard product ensures that these interactions are modeled at an element-wise level, making the network particularly effective at capturing high-order interactions that arise from non-linear transformations applied by the KAN layer.

4.3 Deep Integration of KAN into QResNet: Enhanced Feature Learning

To enhance the integration of KAN into QResNet, spline-based residuals can replace the traditional linear residuals, allowing for more flexible modeling of non-linear interactions. The enhanced QResNet block, incorporating spline-based residuals, is formulated as:

$$\begin{aligned}\mathbf{h}_{\text{Spline-QRes}} &= \sigma(\Phi(W_2 \cdot h_{KAN}) \circ \Phi(W_1 \cdot h_{KAN}) \\ &+\Phi(W_1 \cdot h_{KAN}) + b)\end{aligned} \tag{10}$$

In Eq. (10), $\Phi(\cdot)$ represents the spline transformation applied to the outputs of the linear transformations $W_1\mathbf{h}_{\text{KAN}}$ and $W_2\mathbf{h}_{\text{KAN}}$. The Hadamard product $\circ$ operates on the spline-transformed feature vectors, allowing the model to capture nonlinear interactions in a more flexible and adaptable manner.

This architecture, which incorporates spline-based residuals, provides a more dynamic approach to modeling the underlying data. The use of splines allows the network to better capture the non-linear compositional structures present in the input features, resulting in more accurate predictions in tasks requiring complex feature interactions.

4.4 Modeling Cross-Feature Interactions: Explicit Interaction Modules

The hybrid architecture also includes explicit modeling of cross-feature interactions, which is critical for tasks where interactions between pairs of features are highly predictive of the outcome. These interactions are captured through a combination of KAN and QResNet layers. Pairwise Feature Interactions via QResNet: The interaction between a pair of features i and j is modeled using spline-based transformations and quadratic residuals. This interaction is captured by the following equation:

$$\mathbf{M_{i,j}} = \sigma\left(W_{2,ij} \cdot \left(\phi_i(x_i) \cdot \phi_j(x_j)\right) \circ W_{1,ij} \cdot \left(\phi_i(x_i) \cdot \phi_j(x_j)\right) + W_{1,ij} \cdot \left(\phi_i(x_i) \cdot \phi_j(x_j)\right) + b_{ij}\right) \quad (11)$$

In Eq. (11), the spline-transformed features $\phi_i(x_i)$ and $\phi_j(x_j)$ are multiplied to represent interactions between features i and j. The weight matrices $W_{1,ij}$ and $W_{2,ij}$ are specific to the interaction between features i and j, and b_{ij} is the bias term. The resulting matrix $M_{i,j}$ encapsulates the interaction between features i and j, including both the non-linear transformations from KAN and the quadratic residuals from QResNet.

This formulation allows the model to explicitly capture complex interactions between pairs of features, which is particularly important when these interactions are predictive of the final output. By capturing these interactions, the network can better understand how combinations of features contribute to the overall prediction.

4.5 Aggregation of Interaction Modules

Once the interactions between all feature pairs have been computed, they are aggregated to form a comprehensive feature representation. This aggregated interaction vector $\mathbf{h}_{\text{inter}}$ is obtained by summing over all pairwise interactions:

$$\mathbf{h}_{\text{inter}} = \sum_{i,j} M_{i,j} \quad (12)$$

Eq. (12) represents the aggregation of all pairwise interaction terms into a single vector, $\mathbf{h}_{\text{inter}}$, which encapsulates both the non-linear dependencies and high-order relationships between the input features. This aggregation step is essential for building a comprehensive understanding of how features interact in the input space.

4.6 Final Classification Layer

The final step in the architecture involves passing the aggregated interaction vector $\mathbf{h}_{\text{inter}}$ through a classification layer to produce the output prediction. The output prediction $\hat{\mathbf{y}}$ is computed as:

$$\hat{\mathbf{y}} = \text{softmax}(W_{\text{out}}\, \mathbf{h}_{\text{inter}} + \mathbf{b}_{\text{out}}) \quad (13)$$

In Eq. (13), W_{out} is the weight matrix for the final classification layer, and $\mathbf{b}_{\text{out}}$ is the bias term. The SoftMax function is applied to transform the logits into class probabilities, yielding the final prediction $\hat{\mathbf{y}}$.

The hybrid architecture described in this paper integrates Kolmogorov-Arnold Networks (KAN) and Quadratic Residual Networks (QResNet) to model complex, non-linear interactions between input features. The KAN layer functions as a powerful feature extractor, applying spline transformations to the input data, while the QResNet layers enhance these features by introducing quadratic residuals that capture high-order interactions. In addition, explicit cross-feature interaction modules enable the network to model dependencies between pairs of features, further improving its predictive capabilities.

The mathematical formulations presented in this work, including the spline-based residuals Eq. (10) and the cross-feature interaction modules Eq. (11), demonstrate the power of this hybrid approach in capturing complex feature relationships. By combining these two powerful techniques, the proposed architecture is well-suited for tasks where feature interactions are critical for accurate classification.

V. Experimental Setup

The evaluation of the proposed KAN-enhanced QResNet architecture was conducted using several real-world datasets from e-commerce platforms. These datasets offer rich insights into user behavior and provide diverse scenarios for churn prediction and user behavior modeling. The following subsections outline the datasets, data preprocessing steps, baseline models, model training, and the evaluation metrics used for comparative analysis.

5.1 A. Datasets Description

The experiments utilized three comprehensive e-commerce datasets:

- E-commerce Customer Churn Analysis and Prediction [15]: This dataset includes approximately 5,000 customer entries, featuring customer demographics, purchase histories, and service interactions. It is primarily focused on predicting customer churn, making it suitable for evaluating churn models.

- E-commerce Behavior Data from Multi-Category Store [16]: Comprising over 4 million events, this dataset captures user interactions, including clicks, cart additions, and purchases across multiple categories in an online store. The large size of the dataset allows for testing large-scale interaction predictions.

- E-commerce Customer Churn [17]: Containing approximately 10,000 records, this dataset includes detailed customer interaction logs, transaction histories, and payment details, offering further complexity for churn prediction tasks.

1. E-commerce Customer Churn Analysis and Prediction:

This dataset contains detailed data on a customer-by-customer level an optimum setting for investigating churn. This includes everything from demographics, service interactions and purchase behaviors. Feature selection was performed using methods like correlation matrices to eliminate redundancies and principal component analysis (PCA) to reduce dimensionality without obfuscating key information. They were very customer tenure focused - tenure as a proxy for loyalty. In addition, we generated interaction terms between tenure and purchase frequency to stress on the co-effect of tenure and purchase frequency towards churn.

2. E-commerce Behavior Data from Multi-Category Store:

In essence, human behavior data on over 4 million actions that users perform in different product categories. On account of its volume, feature selection was concentrated to streamlining complexity through the filtering of noise and less informative variables. The top example, the order of operations was around user events like added to and also purchased from cart over just views or clicks. These included session-based features like the average session duration, and time since the previous purchase to encapsulate some temporal dynamics that have shown to be correlated with user engagement or conversion likelihood.

3. E-commerce Customer Churn:

With a fine level of granularity of the first churn dataset, we follow up on this model with specific transaction histories and payment methods necessary to explore churn. Aggregated metrics i.e. total spend per customer, frequency of transactions and average transaction value were created as part of feature engineering phase These features are useful for knowing the difference between a one-time visitor and someone who comes back constantly. We first used the Recursive feature elimination to identify predictors for churn and then we implemented the models using these features.

5.2 Data Preprocessing

All datasets were preprocessed extensively to improve their quality and the model performance. Imputation was made on the missing values and one-hot-encoding for categorical features. Standardizing feature ranges across different datasets for continuous variables. The datasets were split into a training set (80%), validation set (10%), and test set (10%). The large-scale multi-category used manual feature engineering to generate some more meaningful feature interactions, hopefully augmenting the model for performance.

1. Feature Engineering and Selection

Feature engineering and selection are crucial to the performance of a machine learning model. We applied several popular techniques in this study to optimize the model accuracy with computation efficiency on three different datasets. We explain feature engineering and selection steps that were done on each dataset below.

2. E-commerce Customer Churn Analysis and Prediction:

This data includes many customer-level features such as region, age group, total children and their respective ages. Feature engineering: One-Hot Encoding was used for categorical variables like customer region and product type so that the model would be able to utilize these without introducing bias through ordinal relationships. One of the most widely used techniques is One-Hot Encoding where you represent your categorical variables in binary format which helps when you need to include them into machine learning models [39].

RFE was then used to further select the appropriate features. Important: The model works on and removes only one feature at each step based on the worst feature ranking. This method is most helpful while decreasing multicollinearity and luring the pendulum to select features. In the dataset used in this example, customer tenure and total purchase amount were included as they provide excellent predictive power for churn [40], [41].

Mutual Information was also used to measure the relationships between features and the outcome, customer churn. This measure of mutual information is used to capture not only linear dependency but also non-linear dependency that other methods failed to catch. The final model consists of the features with the highest mutual information scores, including purchase frequency and customer engagement level [42].

3. E-commerce Behavior Data from Multi-Category Store:

It contains more than 4 million events: user actions—like clicks, add to cart, and purchases. The feature space also had 46 features, so with only a medium number of samples and the complexity of interactions, Principal Component Analysis (PCA) was used to reduce dimensionality. PCA is applied frequently for the following use cases: Transforms high-dimensional data into a low dimensional subspace so by principal components which account to maximum variance in data are chosen. This process improved the performance of the model while maintaining its good predictive power [43].

Finally, along with PCA, we engineered temporal features like average session duration and time since last purchase based on timestamps to reflect the engagement dynamics of users. These are especially important to model conversion events such as purchases.

In addition, Frequency Encoding was used for high-cardinality categorical features like product category. As the frequency encoding it is computationally cheaper than One-Hot Encoding, especially if we have a lot of categories [44].

4. E-commerce Customer Churn Dataset:

As with the first churn dataset, this version has payment information, historical transactions, and customer segmentation. Feature engineering was primarily interaction features such as total spend per customer and frequency of transactions, from which new metrics were calculated (e. g. average transaction value). These features allowed us to understand customer intents in far more depth and differentiate between first timers and recurring customers.

In these data, feature selection algorithm used: Recursive Feature Elimination (RFE) - it eliminates irrelevant features that have near-zero variance while keeping relevant features such as payment method, and total purchase value. Mutual Information was also used to measure the dependency of customer type and purchase preferences features with target variable (churn) which in turn has helped us keep only most informative features in the model [42].

Feature selection and feature engineering, which were used to improve the predictive performance of the models were conducted across datasets. RFE, Feature Importance by Mutual Information and PCA were used to reduce dimension, drop uncorrelated features and keep only the predictive variables. All these methods were based on extensive published research and have been proven to work across various domains from e-commerce to user behavior prediction.

5.3 Baseline Models

In the experiments, we compared our performance of the KAN-enhanced QResNet model to several baseline models including some higher-order interactions methods to show that it not only generalizes high-order feature interactions but also can capture more complex user behavior. The baseline models included:

- Logistic Regression: A more interpretable, simpler model that is commonly used for binary classification tasks but without the capacity to properly capture non-linear relationships.
- Support Vector Machines (SVMs): It is suitable for high-dimensional spaces, but it can be costly computationally inefficient in the context of large datasets.
- Standard Neural Networks (MLPs): Able to capture non-linearities, but poor performance in representing higher-order interactions without modification.
- QResNet (without KAN layers): Uses quadratic terms to capture higher-order interactions but does not have the decomposition properties of KAN layers for modelling more complex dependencies.

5.4 Model Training

The KAN-enhanced QResNet model was implemented in Python using TensorFlow and trained on a GPU-equipped machine for accelerated computation. The training process involved several steps to ensure the stability and effectiveness of the model:

- Initialization: Weights were initialized with small random values to prevent divergence during early training stages.
- Optimizer: The AdamW optimizer was used to adapt the learning rate dynamically while applying weight decay to prevent overfitting.
- Regularization: L2 regularization and dropout were employed to reduce overfitting, particularly in datasets with a large number of features.
- Hyperparameter Tuning: Key hyperparameters, such as learning rate, batch size, and the number of epochs, were tuned using cross-validation on the validation set.

5.5 Evaluation Metrics

The performance of the KAN-enhanced QResNet and baseline models was evaluated using a set of widely recognized metrics to ensure a comprehensive assessment of model effectiveness. The following metrics were used:

- Accuracy: This metric measures the proportion of correctly classified instances. Given a total of n samples, the accuracy is defined as:

$$\text{Accuracy } = \frac{TP + TN}{TP + TN + FP + FN} \tag{14}$$

where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively.

- Precision: Precision measures the accuracy of positive predictions, particularly in reducing false positives, and is defined as:

$$\text{Precision } = \frac{TP}{TP + FP} \tag{15}$$

- Recall: Recall quantifies how well the model identifies all relevant instances, minimizing false negatives. It is defined as:

$$\text{Recall } = \frac{TP}{TP + FN} \tag{16}$$

- F1-Score: The F1-score is the harmonic mean of precision and recall, providing a balanced measure when both metrics are important. It is given by:

$$\text{F1 } = 2 \times \frac{\text{Precision } \times \text{ Recall}}{\text{Precision } + \text{ Recall}} \tag{17}$$

5.6 Hardware and Software Environment

We performed experiments on a computational environment with an NVIDIA Tesla V100 GPU to enable the requisite computational power in order to train state-of-the-art models on high-dimensional datasets. For model implementation, we use Python 3.8 along with TensorFlow and Scikit-learn as a package for evaluation of the model, pre-processing etc.

Specifically, the experimental setting was devised to test the ability of KAN-enhanced QResNet to outperform traditional ML models particularly in regard to complex, higher-order feature interactions. We evaluate this architecture with a combination of quadratic residuals and KAN layers on different e-commerce datasets to show the use of our approach for real-world dynamic digital ecosystems.

VI. RESULTS

In our experiments, we compared the performance of standard machine learning models such as Logistic Regression and Support Vector Machines (SVM) against sophisticated neural networks like QResNet and KAN-enhanced QResNet architectures in multiple e-commerce datasets. To cover the broad scope of user behavior data complexities and scale the datasets E-Commerce Customer Churn Data, Multi-Category Store Behavior or Customer Churn Analysis were selected. KANs are then compared with standard machine learning models such as logistic regression (LR), SVMs, and conventional neural networks (NN) to show their better predictive performance. They are bench-marked on how accurately they can predict a variety of user actions, such as clicks, purchases, and content consumed from digital platforms. It is believed that KANs will be more efficient than traditional models due to the fact that they can detect hidden, complex and non-linear patterns in user behavior data, otherwise impossible to capture using other algorithms. This superior performance is attributed to the KAN adapted architecture, which allows them to model interactions between variables more effectively, leading to more accurate predictions [45].

The evaluation of these models was based on accuracy, precision, recall, and F_1-score metrics. All results obtained from these datasets are shown in Table I.

6.1 Logistic Regression

Logistic Regression worked fine for binary classification problems that is mostly used for example customer churn problem. It was able to have a test accuracy of 0.75 and an F1-score of 0.68 on the E-Commerce Customer Churn Data. Although its performance deteriorated on more challenging datasets includes Multi-Category Store Behavior, where it achieved a test accuracy of 0.68 and an F1-score of 0.57.

Logistic Regression, on the other hand, seem to perform poorer possibly due to an inherent assumption of linearity. Except this logistic regression has constraint for nonlinear relationship and feature interactions which are widespread phenomena in user Multi-Category Store Behavior data. In Logistic Regression, many features with also complex interpretable interactions could be churned out which are often times useful in common segmented structured linear problems but not the case for Multi-Category Store Behavior with it high dimensionality and multiple non-linear interactions.

6.2 Support Vector Machines (SVM)

Moderate improvements over Logistic Regression were observed with SVM, most significantly in the E-Commerce Customer Churn Data where SVM delivered a test accuracy of 0.80 and an F_1-score of 0.72. However, on Multi-Category Store Behavior dataset the model did not perform well and for test accuracy it was just 0.72 with F_1 score of 0.63.

One of the major strengths of SVM is that it can model non-linear relationships by using kernel functions. Still, the computational complexity of SVM grows with the size of datasets and therefore suffers from the curse of dimensionality (e.g., Multi-Category Store Behavior). SVM is even more than the capable of working with high dimensional data but still too many interactions make it very complex so precisely i think that is why accuracy and F_1-scores are way low than optimal.

6.3 XGBoost

Since XGBoost displayed an authentically superior performance over traditional models, thereby proving to be a robust model across all datasets. It got a test accuracy of 0.84 and an F1-score of 0.77 for E-Commerce Customer Churn data as an example XGBoost is particularly good in capturing the complex, non-linear interactions for its gradient boosting nature when you have weak learners (simple models) consistently, to form a strong predictive model.

XGBoost showed 0.77,0.69 for Test accuracy and F1-score respectively on Multi-Category Store Behaviour We reckon that this is because of the capacity of XGBoost in dealing with high-dimensional data and feature interactions (very common in user behavior prediction). Nevertheless, although XGBoost outperformed traditional models, it was still beaten by more sophisticated architectures such as QResNet, suggesting that there is a need for even larger capacity models to capture higher order interactions.

6.4 Feed Forward Neural Networks (MLP)

The Feed-forward Multilayer Perceptron (MLP) Neural Networks emerged as a top performer especially on the E-Commerce Customer Churn Data, with 0.82 test accuracy and 0.76 F_1-score. Because of their layered structure, neural networks are suited to learn complex non-linear relationships and patterns in the data.

The only exception is in the Multi-Category Store Behavior dataset where MLP's performance went down to a test accuracy of 0.76 and an F1-score of 0.67. Compared to traditional models, the MLP has great flexibility but is not as good at modeling higher-order interactions as QResNet and KAN-enhanced architectures. And further, it is quite possible that the neural network simply overfits in large datasets - without proper regularization this can lead to performance problems with slightly more complex datasets.

6.5 QResNet (No KAN Layers)

QResNet performed good because of its essence to catch the higher-order interactions with quadratic residuals. In the ECommerce Customer Churn Data, QResNet obtained a test accuracy 0.85 and F1-score 0.78 surpassing MLP and XGBoost as well. By modeling quadratic residuals, QResNet was more capable of learning non-linear relationships among the features which helped improve its performance.

QResNet improved test results over Bayesian methods with an accuracy of 0.78 and F1-score of 0.72 on the Multi-Category Store Behavior dataset. It performed less well than the KAN-enhanced QResNet architecture; therefore, while QResNet is able to effectively model non-linear interactions, additional enhancements from KAN layers are required in order to capture complex user behavior in high-dimensional datasets completely.

6.6 KAN-only (No QResNet Layers)

This KAN-only model (without the quadratic residuals due to QResNet), performed quite well, especially on E-Commerce Customer Churn Data, with seven measures such as test accuracy equals 0.80 and F_1-score equals 0.72. But in the Multicategory Store Behavior dataset, our model did worse with test accuracy of 0.74 and F_1-score was 0.65.

Although KAN is good at functional decomposition of high dimensions to univariate components, it cannot model higher order interactions as accurately as QResNet. Since there were no quadratic residuals in this architecture, it might have struggled to capture the full complexity of feature dependencies in the Multi-Category Store Behavior dataset. This indicates that KAN has very good compositional power but still needs improvements to achieve top-tier performance on the larger and more complex datasets as shown by the result of combining it with other improvements in the KAN-enhanced QResNet.

6.7 KAN-enhanced QResNet

The KAN-enhanced QResNet architecture performed in the top of all models on all datasets. The E-Commerce Customer Churn Data, it produced a test accuracy of 0.90 and an F1-score of 0.86 thus showing huge improvement from other models used. This hybrid architecture combines the decomposition power of KAN to model complex interactions and amplify quadratic residuals from QResNet to handle higher-order dependencies.

Moreover, test accuracy on the most difficult dataset (Multi-Category Store Behavior) was also substantial (0.84) with KAN-Enhanced QResNet with F_1-score equals to 0.79. These results showcase robustness of the model — it is able to work with high-dimensional, intricate data where user behavior outcomes depend on many interacting features. By integrating the KAN and QResNet layers, the model can decompose user behavior into simpler parts which help improve model interpretability while still preserving fine-grained feature interactions.

To summarize, this study presents that standard machine learning models like Logistic Regression and SVM are insufficient when dealing with complexity of user behavior in digital ecology, especially on large scale high-dimensional data. Even with improvements brought by XGBoost and MLP, they are still unable to accurately model the intricate higher-order interactions needed for precise predictions.

The QResNet with quadratic residuals (QResNet) significantly improves state-of-the-art and the KAN-enhanced QResNet fantastically enhances this ability. It combines the strength of KAN for composition and QResNet to model higher-order interactions, making it a more accurate and robust solution compared with other models when applied in the challenging task of user behavior prediction over complex digital spaces. For all datasets, the KAN-enhanced QResNet model exhibited the best performance, and we recommend using it for such tasks including customer churn prediction or e-commerce behavior analysis.

VII. Discussion

In this paper, we presented the KAN-enhanced QResNet, a novel hybrid neural architecture designed to push forward user behavior prediction in complex digital ecosystems through an extensive analysis pipeline. It combines the improved interaction modeling of Quadratic Residual Networks (QResNet) with the compositional power of Kolmogorov-Arnold Networks (KAN). These results suggest the utility of advanced architectures that are beyond those established by traditional models; important for high-dimensional, non-linear and dependent datasets commonly seen in modern e-commerce and customer behavior data sets.

Here, we present the good performance of KAN-enhanced QResNet model by empirical results in three different datasets, which are E-Commerce Customer Churn Data, Multi-Category Store Behavior and Customer Churn Analysis. In particular, the hybrid architecture always performed better than all conventional and state-of-the-art models for accuracy, precision, recall as well as F1-score (Logistic Regression, SVM, XGBoost, MLP, QResNet (No KAN Layers)). The highest F_1-scores of the KAN-Enhanced QResNet were 0.86,0.79 on E-Commerce Customer Churn Data and Multi-Category Store Behavior benchmarks respectively, a measure that indicates its ability to capture higher order feature interactions and non-linear dynamics in user behavior.

The KAN-enhanced QResNet has the ability to be a twofold-architecture conscious which is one of its strengths. The KAN layers are particularly good at representing complex functions as a composition of univariate components, which sets the stage for being able to represent the compositional structures within user behavior. This becomes even more beneficial in digital ecosystems, where interactions among different features (example: user actions, demographics, purchase history and so on) are generally compositional in nature. Second, the QResNet layers bring quadratic residuals, which would allow architecture to better capture the second order and higher-order interactions. The decomposition strengths of KAN in concert with the residual learning offered by QResNet lead to the architecture being able to outperform the traditional machine learning methods, which tend to struggle on complicated high-dimensional and massive datasets.

Moreover, the results of this study show that traditional models like Logistic Regression and SVM, which can work on simpler and well-structured problems, do not have enough capacity to be applied to deal with a modern e-commerce data complexity. From here it was identified that the linear assumption of Logistic Regression and the computational inefficiencies in SVM for large datasets, did not allow us to accurately model these complex relationships present in the data. Despite greatly improving the flexibility in handling nonlinear data, both XGBoost and standard neural networks were unable to yield a performance that could overshadow those of QResNet and KAN-integrated enhanced models. Thus, while the KAN-only model is a strong competitor, it is not able to compete in performance with the hybrid model, reconfirming that higher-order interactions must be intentionally modeled for achieving optimal predictive ability.

Perhaps one of the most important lessons to draw from this study is how critical it is for businesses to model feature interactions and non-linear dependencies when they are using digital. For example, e-commerce platforms produce huge amounts of data spread across various dimensions (e.g., user actions, related purchase histories categories of products) which are inherently interrelated by unique constraints. However, traditional models currently in use often fail to correctly capture these nuances, leading to mediocre performance on challenging user behavior prediction tasks like churn prediction and product recommendation.

The performance of the KAN-enhanced QResNets represents a significant step towards overcoming these challenges, demonstrating that quantum learning can indeed generalize across datasets of varied complexity and scale. The consistently superior performance of this hybrid architecture relative to the existing models indicates that it meets the data and other challenges today and will adapt well in future digital ecosystems as we accept that data complexity is a matter of fact.

The results of this study provide conclusive evidence for the significance of advanced hybrid architectures in tackling intricate prediction problems and establishes a new state-of-the-art baseline performance in user behavior prediction. In this paper, we show that things are now possible and largely provide the right direction in both theory and practice for those applications of e-commerce, CRM systems or even further. These findings shed light on where we should be focused with respect to hybrid models, and why this will likely be vital in the face of increasingly complex data environments going forward.

VIII. Conclusion

In this paper, we proposed and rigorously evaluated a hybrid neural network architecture, the KAN-enhanced QResNet, designed to address the complex challenges of user behavior prediction in digital ecosystems. This architecture leverages the compositional capabilities of Kolmogorov-Arnold Networks (KAN) for feature decomposition, paired with the interaction-capturing strengths of Quadratic Residual Networks (QResNet). Our results demonstrate that this hybrid model significantly outperforms traditional machine learning models such as Logistic Regression, SVM, and XGBoost, as well as standard neural networks, across a range of diverse e-commerce datasets.

Empirical evaluations on datasets such as the E-Commerce Customer Churn Data, Multi-Category Store Behavior, and Customer Churn Analysis revealed that the KAN-enhanced QResNet consistently achieves higher predictive accuracy, precision, recall, and F_1-scores. For instance, in the E-Commerce Customer Churn Data, the KAN-enhanced QResNet achieved an F_1-score of 0.86, which is a substantial improvement over traditional models, demonstrating its ability to capture the non-linear, higher-order interactions that are characteristic of complex digital ecosystems.

The contributions of this research are threefold. First, we introduce a novel hybrid architecture that seamlessly integrates KAN's compositional feature extraction capabilities with QResNet's power to model higher-order interactions. Second, we provide a robust empirical analysis across multiple datasets, showcasing the model's adaptability and scalability. Third, we establish the KAN-enhanced QResNet as a state-of-the-art model for user behavior prediction, particularly in scenarios where traditional models struggle to capture intricate feature interactions and non-linear dependencies.

Our findings highlight the limitations of traditional machine learning models, particularly their inability to handle the complex, high-dimensional relationships inherent in modern e-commerce and digital ecosystems. In contrast, the KAN-enhanced QResNet addresses these challenges by effectively modeling both compositional and interaction-based feature dependencies, setting a new benchmark in predictive accuracy for user behavior forecasting. This research not only advances the current state of neural architectures for digital ecosystems but also paves the way for further applications of hybrid models in areas such as personalized recommendations, real-time decision-making, and customer retention strategies.

As far as we are aware, this is the first work that introduced a new structure in neural network design which combines the compositional powers of KAN with interaction modeling properties of QResNet. Such a combination allows the modeling of higher-order dependencies between features and provides a powerful instrument for solving sophisticated prediction problems in digital ecosystems.

Table I: Performance Metrics for Training, Validation, and Test Sets for all Models and Datasets

Model	Dataset	Set	Accuracy	Precision	Recall	F_1-Score
Logistic Regression	E-Commerce Customer Churn Data	Training	0.78	0.75	0.70	0.72
		Validation	0.76	0.73	0.68	0.70
		Test	0.75	0.70	0.65	0.68
	Multi-Category Store Behavior	Training	0.71	0.65	0.60	0.62
		Validation	0.69	0.63	0.58	0.60
		Test	0.68	0.60	0.55	0.57
	Customer Churn Analysis	Training	0.75	0.72	0.65	0.68
		Validation	0.74	0.70	0.62	0.66
		Test	0.73	0.68	0.60	0.64
SVM	E-Commerce Customer Churn Data	Training	0.83	0.80	0.78	0.79
		Validation	0.82	0.78	0.73	0.75
		Test	0.80	0.75	0.70	0.72
	Multi-Category Store Behavior	Training	0.75	0.69	0.63	0.66
		Validation	0.73	0.68	0.61	0.64
		Test	0.72	0.66	0.60	0.63
	Customer Churn Analysis	Training	0.80	0.78	0.72	0.75
		Validation	0.79	0.76	0.70	0.73
		Test	0.78	0.73	0.65	0.69
XGBoost	E-Commerce Customer Churn Data	Training	0.87	0.82	0.80	0.81
		Validation	0.85	0.81	0.78	0.79
		Test	0.84	0.79	0.75	0.77
	Multi-Category Store Behavior	Training	0.80	0.74	0.72	0.73
		Validation	0.79	0.73	0.70	0.72
		Test	0.77	0.71	0.68	0.69
	Customer Churn Analysis	Training	0.84	0.80	0.78	0.79
		Validation	0.82	0.78	0.74	0.76
		Test	0.81	0.75	0.70	0.72
Neural Networks (FF-MLP)	E-Commerce Customer Churn Data	Training	0.85	0.82	0.79	0.80
		Validation	0.83	0.80	0.76	0.78
		Test	0.82	0.78	0.74	0.76
	Multi-Category Store Behavior	Training	0.79	0.75	0.70	0.72
		Validation	0.77	0.73	0.67	0.70
		Test	0.76	0.70	0.65	0.67

Model	**Dataset**	**Set**	**Accuracy**	**Precision**	**Recall**	**F_1-Score**
	Customer Churn Analysis	Training	0.83	0.80	0.75	0.77
		Validation	0.81	0.78	0.72	0.74
		Test	0.80	0.76	0.70	0.73
QResNet (No KAN Layers)	E-Commerce Customer Churn Data	Training	0.88	0.83	0.80	0.82
		Validation	0.87	0.82	0.78	0.80
		Test	0.85	0.80	0.76	0.78
	Multi-Category Store Behavior	Training	0.82	0.78	0.74	0.76
		Validation	0.80	0.76	0.72	0.74
		Test	0.78	0.74	0.70	0.72
	Customer Chum Analysis	Training	0.86	0.81	0.78	0.79
		Validation	0.85	0.80	0.76	0.78
		Test	0.83	0.79	0.74	0.76
KAN only (No QResNet layers)	E-Commerce Customer Churn Data	Training	0.83	0.78	0.75	0.76
		Validation	0.81	0.76	0.72	0.74
		Test	0.80	0.75	0.70	0.72
	Multi-Category Store Behavior	Training	0.77	0.73	0.67	0.70
		Validation	0.75	0.70	0.65	0.67
		Test	0.74	0.69	0.63	0.65
	Customer Chum Analysis	Training	0.81	0.76	0.72	0.74
		Validation	0.80	0.75	0.70	0.72
		Test	0.78	0.73	0.68	0.70
KAN-enhanced QResNet	E-Commerce Customer Churn Data	Training	0.92	0.89	0.87	0.88
		Validation	0.91	0.87	0.85	0.86
		Validation	0.91	0.87	0.85	0.86
		Test	0.90	0.88	0.85	0.86
	Multi-Category Store Behavior	Training	0.88	0.85	0.81	0.83
		Validation	0.86	0.83	0.80	0.81
		Test	0.84	0.80	0.78	0.79
	Customer Churn Analysis	Training	0.90	0.87	0.85	0.86
		Validation	0.89	0.86	0.83	0.85
		Test	0.88	0.85	0.82	0.83

REFERENCES

[1] X. Hu, Y. Yang, S. Zhu, and L. Chen, "Research on a hybrid prediction model for purchase behavior based on logistic regression and support vector machine," in 2020 3rd International Conference on Artificial Intelligence and Big Data (ICAIBD), 2020, pp. 200-204.

[2] K. Wang and Z. Liu, "User behavior prediction using svm," in Proceedings of ICML, 2019.

[3] Q. Zhang, W. Wang, and J. e. a. Li, "Deep learning based social media text-driven stock prediction," IEEE International Conference on Big Data, 2018.

[4] X. Li and Y. Tang, "A study on logistic regression for customer behavior prediction," Journal of Data Science, vol. 57, pp. 233-245, 2019.

[5] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 785-794.

[6] P. Sun and H. Wang, "A user purchase behavior prediction method based on xgboost," IEEE Access, 2019.

[7] N. Karps and E. Johansson, "Kolmogorov-arnold networks in predictive modeling: Applications and challenges," Journal of Machine Learning Research, vol. 23, pp. 1-19, 2021.

[8] I. Goodfellow, Y. Bengio, and A. Courville, Deep learning. MIT press, 2016.

[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning in practice: Neural networks in behavioral analytics," Nature, vol. 521, pp. 436-444, 2015.

[10] J. H. Friedman, "An improved random forest approach for customer behavior prediction," Journal of Statistical Learning, 2015.

[11] L. Xu and Q. Ma, "Random forests in behavior prediction: An empirical evaluation," Data Mining and Knowledge Discovery, 2021.

[12] V. I. Arnold, "On the representation of continuous functions by sums of univariate functions," Doklady Akademii Nauk, 1957.

[13] Z. Liu and Y. Wang, "Kan 2.0: Kolmogorov-arnold networks meet science," IEEE Transactions on Neural Networks, 2024.

[14] J. Bu and A. Karpatne, "Quadratic residual networks for solving forward and inverse problems in physics," SIAM Journal of Computing, 2021.

[15] A. Verma, "Ecommerce customer churn analysis and prediction," 2010, accessed: September 22, 2024. [Online]. Available: https: //www.kaggle.com/datasets/ankitverma2010/ecommerce-customer-churn-analysis-and-prediction

[16] M. Kechinov, "Ecommerce behavior data from multi-category store," 2020, accessed: September 22, 2024. [Online]. Available: https: //www.kaggle.com/datasets/mkechinov/ecommerce-behavior-data-from-multi-category-store

[17] S. Semaya, "E-commerce customer churn," 2020, accessed: September 22, 2024. [Online]. Available: https://www.kaggle.
com/datasets/samuelsemaya/ e-commerce-customer-churn/data

[18] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara, "Variational autoencoders for collaborative filtering," Proceedings of the 2018 World Wide Web Conference, pp. 689-698, 2018.

[19] R. Baeza-Yates, "User behavior analytics: A survey," IEEE Internet Computing, vol. 22, no. 4, pp. 68-76, 2018.

[20] Y. Zhou and Y. Koren, "Deep learning for recommender systems: A Netflix case study," in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018, pp. 197-206.

[21] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," arXiv preprint arXiv:1511.06939, 2016.

[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[23] Z. C. Lipton, "The mythos of model interpretability," arXiv preprint arXiv:1606.03490, 2016.

[24] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," Nature Machine Intelligence, vol. 1, no. 5, pp. 206-215, 2019.

[25] A. N. Kolmogorov, "On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition," Dokl. Akad. Nauk SSSR, vol. 114, pp. 953-956, 1957.

[26] V. I. Arnold, "On functions of three variables," Collected Works: Representations of Functions, Celestial Mechanics and KAM Theory, 1957-1965, pp. 5-8, 1963.

[27] V. Kurkova, "Kolmogorov's theorem and multilayer neural networks," Neural networks, vol. 5, no. 3, pp. 501-506, 1992.

[28] T. Poggio and F. Girosi, "Networks for approximation and learning," Proceedings of the IEEE, vol. 78, no. 9, pp. 1481-1497, 1990.

[29] J. Braun and M. Griebel, "Kolmogorov's theorem and multilevel approximation of multivariate functions," Journal of Approximation Theory, vol. 159, no. 1, pp. 39-57, 2009.

[30] G. Montúfar, R. Pascanu, K. Cho, and Y. Bengio, "On the number of linear regions of deep neural networks," in Advances in neural information processing systems, vol. 27, 2014.

[31] C. Ma, P. Kang, and X. Liu, "Hierarchical gating networks for sequential recommendation," in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 825-833.

[32] C. A. Gomez-Uribe and N. Hunt, "Netflix recommendations: Beyond the 5 stars (part 1)," The Netflix Tech Blog, 2015.

[33] M. McCarthy, "How Spotify's discover weekly cracked human curation at internet scale," The Verge, 2016.

[34] Q. V. Le, V. T. Nguyen, and N. T. Thoai, "Machine learning in online advertising," Journal of Marketing Research, vol. 57, no. 3, pp. 442-464, 2020.

[35] H. Xu, X. Luo, J. M. Carroll, and M. B. Rosson, "Personalized online advertising: An information processing theory approach," Journal of Marketing Research, vol. 51, no. 5, pp. 672-686, 2014.

[36] M. Huang, W. Wu, Q. Tang, and J. Yu, "Personalized user experience on e-commerce platforms: A machine learning approach," International Journal of Machine Learning and Cybernetics, vol. 10, no. 1, pp. 181-197, 2019.

[37] L. A. Leiva, M. Albrecht, and A. Alizadeh, "Interaction optimization in e-commerce websites: An empirical study on user experience and business outcomes," Journal of Interaction Science, vol. 7, no. 1, pp. 1-14, 2019.

[38] J. Li, H. Yu, Q. Wang, and W. Liu, "Deep learning for personalized user experience in mobile applications," IEEE Transactions on Mobile Computing, vol. 19, no. 6, pp. 1233-1245, 2020.

[39] D. Kim et al., "One-hot encoding-based feature engineering for predicting the importance of academic citations," Journal of Data Science, pp. 325-334, 2020.

[40] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," Machine Learning, vol. 46, no. 1, pp. 389-422, 2002.

[41] S. Oh et al., "Hybrid-recursive feature elimination for efficient feature selection," Applied Sciences, vol. 10, no. 9, p. 3211,2020.

[42] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 8, pp. 1226-1238, 2005.

[43] I. Jolliffe, Principal component analysis. Springer, 2011.

[44] R. Mishra, P. Pundir, et al., "Comparison of categorical variable encoding techniques for neural network classifiers," International Journal of Computer Applications, vol. 182, no. 44, 2018.

[45] K. Yin and X. Tan, "Convolutional Kolmogorov-Arnold networks," in Proceedings of the 31st IEEE Conference on Computer Vision and Pattern Recognition (CVPR) IEEE, 2024, pp. 123-134.