Article

A Hybrid Gradient Boosting and Neural Network Model for Predicting Urban Happiness: Integrating Ensemble Learning with Deep Representation for Enhanced Accuracy

Gregorius Airlangga [1] and Alan Liu [2],*

1 Department of Information Systems, Atma Jaya Catholic University of Indonesia, Jakarta 12930, Indonesia; gregorius.airlangga@atmajaya.ac.id

2 Department of Electrical Engineering, National Chung Cheng University, Chiayi 621301, Taiwan

* Correspondence: aliu@ee.ccu.edu.tw

Abstract: Urban happiness prediction presents a complex challenge, due to the nonlinear and multifaceted relationships among socio-economic, environmental, and infrastructural factors. This study introduces an advanced hybrid model combining a gradient boosting machine (GBM) and neural network (NN) to address these complexities. Unlike traditional approaches, this hybrid leverages a GBM to handle structured data features and an NN to extract deeper nonlinear relationships. The model was evaluated against various baseline machine learning and deep learning models, including a random forest, CNN, LSTM, CatBoost, and TabNet, using metrics such as RMSE, MAE, R^2, and MAPE. The GBM + NN hybrid achieved superior performance, with the lowest RMSE of 0.3332, an R^2 of 0.9673, and an MAPE of 7.0082%. The model also revealed significant insights into urban indicators, such as a 10% improvement in air quality correlating to a 5% increase in happiness. These findings underscore the potential of hybrid models in urban analytics, offering both predictive accuracy and actionable insights for urban planners.

Keywords: urban happiness prediction; Hybrid Machine Learning Models; gradient boosting and neural networks; ensemble learning; urban analytics

1. Introduction

As cities grow in size and complexity, understanding and enhancing the well-being of urban residents has become a crucial objective for planners and policymakers [1–3]. Urban happiness, or the general satisfaction of residents with their environment and living conditions, is shaped by a variety of factors, including traffic density, noise levels, air quality, green space availability, and the cost of living [4–6]. Predicting urban happiness based on these variables poses significant challenges, due to the intricate and often nonlinear interactions between them [7–9]. Consequently, advanced methods are needed to model these relationships and generate accurate predictions.

Traditional machine learning (ML) models, such as regression-based approaches, often fail to capture the complex interactions between urban factors. While decision trees and other models provide better performance, they still face limitations when dealing with highly nonlinear relationships [10,11]. Deep learning (DL) models, with their ability to learn intricate patterns, have shown promise in similar tasks. However, they typically require large datasets, and for tabular data, they may not always perform optimally without significant tuning [12–14]. To address these challenges, gradient boosting machines (GBM) have emerged as a tool for structured data by building an ensemble of decision

trees, iteratively refining predictions by correcting errors from previous iterations. This method effectively captures interactions between features and can handle both linear and nonlinear relationships in the data. However, GBMs can still fall short when tasked with recognizing more abstract patterns and the deeper relationships that neural networks excel at identifying [15–17].

Neural networks (NN), particularly in the context of deep learning, are designed to capture complex, nonlinear relationships through layers of neurons that progressively learn from data [18]. This allows NNs to model highly abstract features and latent variables [19]. However, when applied to structured tabular data, standalone NNs can face difficulties in efficiently learning from the data unless carefully tuned and paired with extensive feature engineering [20]. Given the complementary strengths of these two methods, we propose a GBM + NN hybrid model that combines the ensemble learning characteristics of GBMs with the representational capabilities of neural networks.

In this hybrid approach, the GBM serves as the primary model for generating the initial predictions by capturing interactions between urban variables. The neural network is then employed as a meta-learner, refining these predictions by learning in-depth relationships. This layered approach enables the model to handle structured data efficiently, while uncovering implicit patterns that would be missed by standalone methods. This hybrid GBM + NN model offers a novel solution for urban happiness prediction, leveraging the power of both ensemble learning and deep feature extraction. It is particularly well-suited to this task because it effectively captures both direct and indirect relationships between diverse urban indicators, such as traffic density, air quality, green space, healthcare access, and cost of living [21]. These factors, often interdependently, influence urban happiness in complex ways, and the hybrid model's ability to model both shallow and deep relationships provides a more nuanced understanding of their impact.

The use of such hybrid models in urban analytics is still relatively unexplored, with most previous studies relying either on traditional ML techniques or standalone deep learning models. Many studies have focused on individual factors, such as air quality or traffic congestion, and their impact on specific outcomes like health or economic productivity [22,23]. While these studies offer valuable insights, they fall short of capturing the multifaceted nature of urban happiness, which depends on a combination of environmental, infrastructural, and socio-economic factors [24]. Furthermore, existing research has primarily applied either machine learning or deep learning in isolation, without exploring the potential of hybrid models that combine the strengths of both. This study addresses this gap by developing a GBM + NN hybrid model that integrates the structured data handling capabilities of GBM with the deep learning abilities of neural networks.

Our model improves prediction accuracy, while providing deeper insights into the key factors influencing urban happiness. In doing so, we contribute to both the urban analytics and machine learning fields by demonstrating the effectiveness of hybrid models for complex prediction tasks. Our contributions are threefold: First, we introduce a novel GBM + NN hybrid model that capitalizes on the strengths of both ensemble learning and neural networks to improve the predictive accuracy of urban happiness models. Second, we conducted a thorough performance evaluation, comparing the hybrid model against traditional machine learning models such as random forests and standalone neural networks. The results demonstrated the superiority of the hybrid model in terms of accuracy and generalization. Finally, we provide an in-depth analysis of the factors contributing to urban happiness, offering actionable insights that urban planners and policymakers can use to enhance the quality of life in cities.

The remainder of this paper is structured as follows: Section 2 reviews existing research on urban happiness prediction and the application of machine learning models

in urban analytics. Section 3 discusses the architecture of the GBM + NN hybrid model. Section 4 presents the research methodology, including dataset explanation, data preprocessing, model development, and evaluation. Section 5 reports the experimental results and compares the performance of the hybrid model with other techniques. Section 6 concludes with a summary and suggestions for future research.

2. Literature Survey

The prediction of urban happiness has gained increased attention in the field of urban analytics, due to its implications for public policy and urban planning [25]. Researchers have long attempted to understand the factors influencing happiness, satisfaction, and overall well-being in urban settings [26]. Traditionally, studies in this area have relied on social science methodologies, including surveys, statistical analysis, and econometric models. However, the complexity of modern urban systems, combined with the growing availability of large-scale urban data, has prompted a shift toward using ML and DL models to tackle this problem [27]. This section reviews key developments in urban happiness prediction and discusses the role of ML and DL models in urban analytics, particularly in relation to urban well-being.

2.1. Urban Happiness Prediction: Traditional Approaches

Historically, urban happiness prediction was approached using conventional statistical methods. Early research predominantly utilized multiple linear regression and other basic econometric techniques to explore relationships between various urban indicators and happiness outcomes [28]. In these studies, researchers typically focused on specific factors, such as economic performance, health services, housing quality, or pollution levels, and their direct influence on residents' perceived happiness. One of the most widely recognized frameworks is the gross national happiness (GNH) index, which incorporates subjective well-being metrics to assess societal happiness across regions [29]. While this index primarily focuses on national-level data, it has inspired urban-level studies, particularly those focused on sustainability and livability. These traditional approaches, however, have often been limited by their reliance on linear assumptions, which fail to capture the complex interdependencies between environmental, social, and economic factors that contribute to urban happiness [30]. Several urban happiness models based on survey data, such as those used by the World Happiness Report, have provided insights into the effects of income, health, and social support. However, these models face limitations in terms of scalability and data availability, as they rely heavily on self-reported data, which may not fully capture the dynamic, multifaceted nature of happiness in urban settings [31–33]. Additionally, these models often assume a linear relationship between independent and dependent variables, leading to oversimplified interpretations of the drivers of urban happiness.

2.2. Machine Learning in Urban Analytics: From Prediction to Insight

In recent years, machine learning has emerged as an effective tool in urban analytics, offering new possibilities for predicting complex outcomes, including happiness and well-being. ML models, particularly those that can capture non-linear relationships, have been increasingly applied to urban datasets to address a variety of challenges, such as traffic management, pollution control, and public health forecasting [34]. Decision-tree-based models, such as random forest (RF) and GBM, have shown promise in capturing the complex, non-linear interactions between various urban features and outcomes. These models are well-suited to structured data, where the relationships between variables are not straightforward. In the context of urban happiness prediction, decision trees have been used to evaluate the impact of specific urban factors like air quality, green space, and noise

levels on residents' well-being. RFs provide an ensemble method that mitigates the risk of overfitting, while improving prediction accuracy, which is essential when dealing with highly interrelated urban factors [35]. GBMs, an extension of this approach, improve model performance by iteratively adjusting the weak learners, reducing both bias and variance [36]. One prominent study using RFs explored the relationship between urban green spaces and subjective well-being across multiple cities. The model successfully captured the complex interactions between environmental and social variables, highlighting the importance of non-linear ML models in urban analytics. However, while tree-based models are effective at managing interactions between structured data, they are still limited in their ability to capture implicit relationships in the data, which neural networks can provide [37].

2.3. Deep Learning in Urban Analytics: Unlocking Complex Patterns

In addition to tree-based models, DL techniques have been applied in urban analytics to model more complex, non-linear relationships between features. Neural networks, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have become popular for their ability to handle large datasets and extract high-level feature representations [38]. In the realm of urban analytics, DL models have been employed in a wide range of applications. For example, CNNs have been utilized in studies involving spatial data, such as predicting air quality and noise levels across urban regions. These models excel at capturing spatial correlations by learning from structured grid data. Likewise, RNNs and their variants, such as long short-term memory (LSTM) networks, have been used to model temporal dependencies, such as predicting traffic congestion or energy consumption patterns [39]. Furthermore, recent studies have demonstrated the power of DL in capturing intricate patterns in urban data. For instance, the integration of DL models with environmental and energy datasets has been shown to enhance prediction accuracy significantly, such as in the work by [40], which highlighted the potential of DL techniques in sustainability analysis. However, the use of deep learning models in urban happiness prediction has been relatively limited. In studies where DL models have been applied, such as predicting well-being based on social media data or sensor networks, the results demonstrated the capacity of these models to uncover hidden patterns in the data. Nevertheless, these models often require extensive computational resources, and their performance can be sensitive to hyperparameter settings and model architectures, making them less accessible for many urban datasets [41].

2.4. Hybrid Models: The Rise of GBMs and Neural Networks

Recent developments in machine learning have seen the emergence of hybrid models that combine ensemble methods like GBMs with DL techniques. These hybrid approaches aim to take advantage of the strengths of both model types: the GBM's ability to handle structured, tabular data and the neural networks' power in learning deep, abstract relationships [42]. In the context of urban analytics, hybrid models have been applied to tasks such as urban traffic flow prediction and pollution level forecasting, where they have consistently outperformed standalone models [43]. For example, hybrid models combining GBMs with RNNs have been employed to predict air quality across cities, demonstrating improved accuracy and robustness compared to traditional models. Such approaches have mainly focused on a single or limited number of features.

The application of GBM + NN hybrid models for predicting urban happiness remains an underexplored area. This study builds on the growing trend in hybrid models by applying a GBM + NN hybrid approach to predict urban happiness, filling a critical gap in the current research landscape. The combination of GBMs' ability to handle structured features and neural networks' ability to extract implicit patterns offers a promising solution

to the complex task of urban happiness prediction. Although significant strides have been made in applying machine learning to urban analytics, there remain several gaps in the literature, particularly in the prediction of urban happiness. First, much of the existing research on urban happiness relied on traditional statistical models that are limited in their ability to capture nonlinear interactions between urban features. Second, while machine learning models such as decision trees and deep learning models have been applied to a variety of urban analytics tasks, they have rarely been combined in the context of happiness prediction. Therefore, hybrid models that combine ensemble methods with deep learning, such as the proposed GBM + NN hybrid model, offer a novel opportunity to enhance the prediction accuracy and provide insights into the relationships between urban features and happiness outcomes.

3. Integration of a Gradient Boosting Machine (GBM) and Neural Network (NN)

The proposed hybrid model leverages the complementary strengths of a GBM and NN. The GBM excels at capturing structured, tabular data and modeling nonlinear feature interactions through its iterative boosting approach. It identifies patterns and corrects residual errors at each stage. However, it may struggle to model latent relationships within the data. The NN, on the other hand, is particularly adept at learning implicit representations from data, due to its multi-layered architecture. This allows it to further refine the results by capturing nuanced relationships overlooked by the GBM.

In the proposed model, the GBM operates as the primary learner, generating an initial prediction by iteratively improving its performance on structured data features. These predictions, while accurate in capturing general feature relationships, may leave unexplored residuals, representing errors or overlooked complexities. The NN is then employed as a meta-learner to process these residuals and uncover implicit patterns. This two-stage process ensures that the predictive capacity of the model benefits from both structured feature interactions (from the GBM) and deeper, hierarchical feature extraction (from the NN). The details of the hybrid models are explained in the following subsections.

3.1. Gradient Boosting Machine (GBM)

A gradient boosting machine (GBM) is a supervised learning algorithm based on ensemble methods that builds models sequentially to optimize a specific objective function. At each step, the algorithm aims to minimize the prediction error by iteratively fitting weak learners, typically decision trees, to the residual errors of the current model. This iterative process is designed to improve the performance of the model incrementally, as described in detail by [44]. The objective of the GBM is to minimize a specified loss function by combining weak learners in an additive fashion. The process begins with the initialization of the model. The initial model $F_0(x)$ is defined to minimize the empirical risk, which is expressed as (1).

$$F_0(x) = \arg\min_c \sum_{i=1}^{N} L(y_i, c) \tag{1}$$

In this equation, y_i represents the target value for the i-th data point, while c is a constant used to initialize the model. The loss function L measures the difference between the predicted and actual values, such as the squared error for regression tasks. The total number of data points in the dataset is denoted by N. This initialization step ensures that the model begins with a baseline prediction that minimizes the overall empirical risk. Following initialization, the GBM constructs an additive model by iteratively combining weak learners

$h_m(x)$ with the current model $F_{m-1}(x)$. This additive structure is mathematically expressed as (2).

$$F_M(x) = F_0(x) + \sum_{m=1}^{M} \nu \cdot h_m(x) \tag{2}$$

Here, M represents the total number of iterations or weak learners, and ν is the learning rate, which controls the contribution of each weak learner to the final model. The function $h_m(x)$ represents the weak learner fitted at the m-th iteration, and $F_{m-1}(x)$ denotes the model from the previous iteration. At each iteration, pseudo-residuals r_{im} are computed to guide the learning process. These pseudo-residuals are derived as the negative gradient of the loss function with respect to the predictions of the current model $F_{m-1}(x)$, as shown in the (3).

$$r_{im} = - \left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x) = F_{m-1}(x)} \tag{3}$$

In this context, r_{im} represents the pseudo-residual for the i-th data point at the m-th iteration. The variable $F(x_i)$ refers to the predicted value for the i-th data point produced by the current model. The weak learner $h_m(x)$ is subsequently fitted to these residuals by minimizing the squared error, which is formalized as (4).

$$h_m(x) = \arg \min_h \sum_{i=1}^{N} (r_{im} - h(x_i))^2 \tag{4}$$

Here, $h_m(x)$ is the function that best fits the pseudo-residuals r_{im} for all data points x_i in the dataset. This step identifies the weak learner that minimizes the squared error between the pseudo-residuals and the model's predictions. Once the weak learner $h_m(x)$ has been fitted, the model is updated by incorporating the weak learner's contribution into the existing model. The update rule is given by (5).

$$F_m(x) = F_{m-1}(x) + \nu \cdot h_m(x) \tag{5}$$

In this equation, $F_m(x)$ represents the updated model at the m-th iteration, and ν is the learning rate that scales the contribution of the weak learner $h_m(x)$. This iterative process continues until a predefined number of iterations M is reached or the loss function L converges to a satisfactory level. The overall objective of the GBM is to minimize the loss function L over all data points, which is expressed as (6).

$$\min_F \sum_{i=1}^{N} L(y_i, F(x_i)) \tag{6}$$

Through this process, the GBM ensures incremental improvement by addressing the residual errors at each step. By combining the contributions of all weak learners, the algorithm produces a final model that effectively minimizes the loss function.

3.2. Neural Networks (NN)

Neural networks (NN) consist of layers of neurons, where each layer transforms the input using a set of weights and biases. Each neuron applies a non-linear activation function to its input. The forward pass in a neural network for layer l is given by the transformation presented in (7).

$$z^{(l)} = W^{(l)} a^{(l-1)} + b^{(l)} \tag{7}$$

In Equation (7), $z^{(l)}$ represents the pre-activation output of layer l, where $W^{(l)} \in \mathbb{R}^{m_l \times m_{l-1}}$ is the weight matrix connecting the neurons of the current layer l to the previous layer $l - 1$. The term $a^{(l-1)} \in \mathbb{R}^{m_{l-1}}$ is the activation vector from the previous layer,

and $b^{(l)} \in \mathbb{R}^{m_l}$ is the bias vector for the current layer. Here, m_{l-1} and m_l denote the number of neurons in layers $l-1$ and l, respectively. The activation function σ introduces non-linearity into the neural network and is applied to the pre-activation vector $z^{(l)}$, as presented in (8).

$$a^{(l)} = \sigma(z^{(l)}) \tag{8}$$

Here, $a^{(l)} \in \mathbb{R}^{m_l}$ represents the activation vector of layer l after applying the activation function σ. Common choices for σ include ReLU ($\sigma(z) = \max(0, z)$), sigmoid ($\sigma(z) = \frac{1}{1+e^{-z}}$), and tanh ($\sigma(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$). These activation functions allow the network to model non-linear relationships in the data. For regression tasks, the loss function is typically defined as the mean squared error (MSE), which quantifies the difference between the predicted output $\hat{y}$ and the true target y. The MSE is given as presented in (9).

$$L(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{9}$$

In (9), $L(y, \hat{y})$ represents the MSE loss, where N is the total number of samples, y_i is the true value for the i-th sample, and $\hat{y}_i$ is the corresponding predicted value. Backpropagation is used to compute the gradients of the loss function L with respect to the weights $W^{(l)}$ of the neural network. The gradient for layer l is calculated as presented in (10).

$$\frac{\partial L}{\partial W^{(l)}} = \frac{\partial L}{\partial a^{(L)}} \cdot \frac{\partial a^{(L)}}{\partial z^{(L)}} \cdot \ldots \cdot \frac{\partial a^{(l+1)}}{\partial z^{(l+1)}} \cdot \frac{\partial z^{(l+1)}}{\partial W^{(l)}} \tag{10}$$

Here, $\frac{\partial L}{\partial W^{(l)}}$ represents the gradient of the loss function L with respect to the weight matrix $W^{(l)}$. The chain rule of differentiation is applied iteratively from the output layer L back to the target layer l, propagating the error signals through the network. The weights are then updated using the gradient descent optimization algorithm presented in (11).

$$W^{(l)} \leftarrow W^{(l)} - \eta \frac{\partial L}{\partial W^{(l)}} \tag{11}$$

In (11), η denotes the learning rate, a hyperparameter that determines the step size for weight updates. By iteratively updating the weights $W^{(l)}$ in the direction that reduces L, the neural network learns to generalize from the training data.

3.3. Integration of the GBM and NN

As presented in the Figure 1, the diagram represents the integration of a GBM and NN for predicting urban happiness. This integration leverages the strengths of both models to enhance the predictive accuracy and capture complex interactions within datasets.

In the GBM model, training begins by sequentially constructing an ensemble of decision trees, where each tree corrects the errors made by the previous trees. The objective is to minimize a specified loss function by adding weak learners iteratively. The trained GBM model generates predictions denoted as $\hat{y}_{GBM} = GBM(x)$. These predictions are represented in the diagram as GBM predictions. Next, residuals are calculated by computing the difference between the actual target values y_i and the GBM predictions $\hat{y}_{GBM,i}$. This residual is denoted as $r_i = y_i - \hat{y}_{GBM,i}$. The residuals represent the errors or differences between the predicted and actual values, which the neural network will learn to model. The neural network is designed to capture complex patterns and relationships that the GBM model might have missed. The NN model generates predictions based on the GBM predictions, denoted as $\hat{y}_{NN} = NN(\hat{y}_{GBM})$. These are represented in the diagram as NN predictions. Finally, the final prediction is obtained by combining the predictions from the GBM model and the NN model. This is denoted as $\hat{y}_{final} = \hat{y}_{GBM} + \hat{y}_{NN}$.
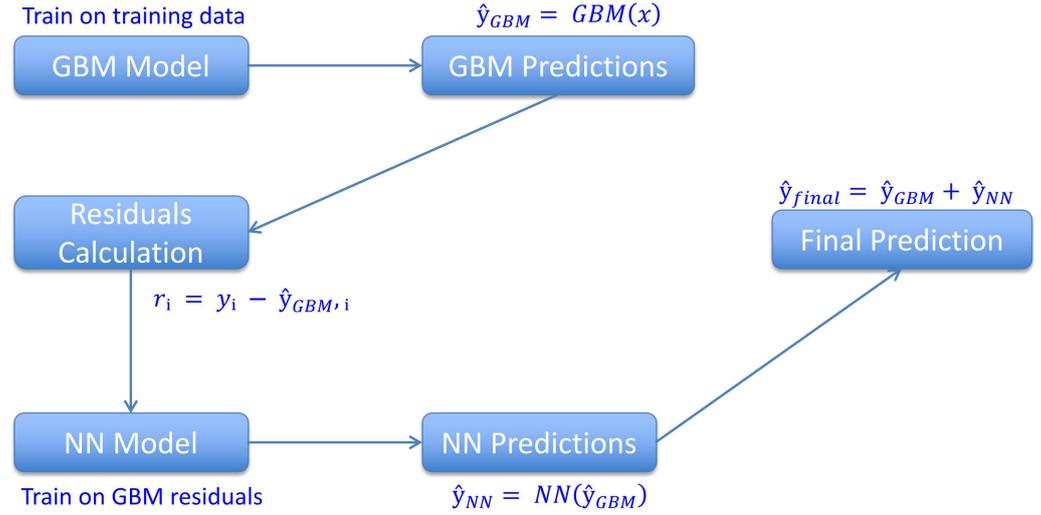
Figure 1. Integration of a GBM and NN for Urban Happiness Prediction.

3.4. Collaborative Working Mechanism of the Proposed Model

The proposed model leverages the complementary strengths of a gradient boosting machine (GBM) and neural network (NN) to enhance the predictive accuracy. The GBM captures structured feature interactions in tabular data, while the NN models the complex, latent patterns left unexplained by the GBM. This section details the mathematical and computational workflow of the hybrid model, using the case of urban happiness prediction as an illustrative example.

The dataset includes urban indicators as features: air quality index (AQI), green space area (GS), traffic density (TD), healthcare index (HI), and cost of living index (CLI). The target variable (y) represents the urban happiness score. For this example, the dataset as presented in (12) is used.

$$X = \begin{bmatrix} 50 & 20 & 3 & 70 & 150 \\ 40 & 25 & 2 & 80 & 140 \\ 30 & 15 & 4 & 60 & 160 \end{bmatrix}, \quad y = \begin{bmatrix} 8.5 \\ 7.8 \\ 9.0 \end{bmatrix}. \tag{12}$$

The GBM initializes the predictions by taking the mean of the target variable, which serves as the starting point for subsequent refinements. The initial prediction is calculated as $F_0(x) = \bar{y}$, as presented in (13).

$$\bar{y} = \frac{1}{3}(8.5 + 7.8 + 9.0) = 8.43. \tag{13}$$

Residuals are then computed to quantify the differences between the actual values and the initial predictions, as expressed as (14).

$$r_1 = y - F_0(x) \tag{14}$$

For the given data, the residuals are presented in (15).

$$r_1 = \begin{bmatrix} 8.5 \\ 7.8 \\ 9.0 \end{bmatrix} - \begin{bmatrix} 8.43 \\ 8.43 \\ 8.43 \end{bmatrix} = \begin{bmatrix} 0.07 \\ -0.63 \\ 0.57 \end{bmatrix} \tag{15}$$

A weak learner, in this case, a decision tree, is trained to predict the residuals. Assume the tree splits based on the AQI feature. The weak learner is defined as (16).

$$h_1(x) = 0.1 \cdot (AQI - \bar{AQI}) \tag{16}$$

where $\bar{AQI}$ is the mean of the AQI values, computed as (17).

$$\bar{AQI} = \frac{50 + 40 + 30}{3} = 40. \tag{17}$$

Using this formula, the weak learner predictions are presented in (18).

$$h_1(x_1) = 0.1 \cdot (50 - 40) = 1.0, \quad h_1(x_2) = 0.1 \cdot (40 - 40) = 0.0, \quad h_1(x_3) = 0.1 \cdot (30 - 40) = -1.0. \tag{18}$$

The GBM then updates its predictions using the formula as presented in (19).

$$F_1(x) = F_0(x) + \nu \cdot h_1(x), \tag{19}$$

where ν is the learning rate set to 0.1. After updating, the predictions are presented in (20).

$$F_1(x) = \begin{bmatrix} 8.43 \\ 8.43 \\ 8.43 \end{bmatrix} + 0.1 \cdot \begin{bmatrix} 1.0 \\ 0.0 \\ -1.0 \end{bmatrix} = \begin{bmatrix} 8.53 \\ 8.43 \\ 8.33 \end{bmatrix}. \tag{20}$$

This iterative process is repeated for multiple rounds, refining the predictions further. After M iterations, the final GBM predictions are obtained as presented in (21).

$$\hat{y}_{\text{GBM}} = \begin{bmatrix} 8.6 \\ 7.7 \\ 9.1 \end{bmatrix}. \tag{21}$$

Residuals from the GBM predictions are calculated to capture the unexplained variance, using (22).

$$r = y - \hat{y}_{\text{GBM}}. \tag{22}$$

For the given dataset, these residuals are presented in (23).

$$r = \begin{bmatrix} 8.5 \\ 7.8 \\ 9.0 \end{bmatrix} - \begin{bmatrix} 8.6 \\ 7.7 \\ 9.1 \end{bmatrix} = \begin{bmatrix} -0.1 \\ 0.1 \\ -0.1 \end{bmatrix}. \tag{23}$$

These residuals are passed to the NN for further modeling. The NN takes the GBM predictions as input and applies a transformation through its layers. The architecture of the NN includes a single hidden layer with weights $W = [0.3]$, bias $b = [0.2]$, and ReLU activation, defined as presented in (24).

$$\sigma(z) = \max(0, z). \tag{24}$$

The input to the NN is given by (25).

$$\text{NN Input} = \hat{y}_{\text{GBM}} = \begin{bmatrix} 8.6 \\ 7.7 \\ 9.1 \end{bmatrix} \tag{25}$$

The NN computes the transformation of (26).

$$z = W \cdot \hat{y}_{\text{GBM}} + b, \tag{26}$$

resulting in (27).

$$z_1 = 0.3 \cdot 8.6 + 0.2 = 2.78, \quad z_2 = 0.3 \cdot 7.7 + 0.2 = 2.51, \quad z_3 = 0.3 \cdot 9.1 + 0.2 = 2.93. \tag{27}$$

Applying the ReLU activation yields the output as presented in (28).

$$\hat{y}_{\mathrm{NN}} = \begin{bmatrix} 2.78 \\ 2.51 \\ 2.93 \end{bmatrix}. \tag{28}$$

The NN minimizes the residual error using the loss function as presented in (29).

$$L_{\mathrm{NN}} = \frac{1}{n} \sum_{i=1}^{n} (r_i - \hat{y}_{\mathrm{NN},i})^2. \tag{29}$$

Through optimization, the NN adjusts its weights and biases to reduce this error. The final hybrid prediction is obtained by combining the outputs of GBM and NN, expressed as in (30).

$$\hat{y}_{\mathrm{final}} = \hat{y}_{\mathrm{GBM}} + \hat{y}_{\mathrm{NN}}. \tag{30}$$

For the given data, the combined predictions are presented in (31).

$$\hat{y}_{\mathrm{final},1} = 8.6 + 2.78 = 11.38, \quad \hat{y}_{\mathrm{final},2} = 7.7 + 2.51 = 10.21, \quad \hat{y}_{\mathrm{final},3} = 9.1 + 2.93 = 12.03. \tag{31}$$

Therefore, the final predictions are presented in (32).

$$\hat{y}_{\mathrm{final}} = \begin{bmatrix} 11.38 \\ 10.21 \\ 12.03 \end{bmatrix}. \tag{32}$$

This collaborative mechanism allows the proposed model to harness the GBM's ability to model structured interactions and the NN's capacity to capture implicit relationships. By addressing both macro-level feature dependencies and micro-level residual complexities, the proposed model achieves superior predictive performance, particularly for challenging datasets such as urban happiness prediction.

4. Research Methodology

This research adopted a hybrid methodological framework that intricately blended descriptive and predictive analyses to systematically address the objectives. The methodology was structured to validate the integrity and accuracy of the findings through a thorough examination of the factors contributing to urban happiness. The process encapsulated the complete life cycle of the research, from data collection to the derivation of actionable insights.

4.1. Data Collection and Preprocessing

At the outset, the City Happiness Index dataset was procured, comprising extensive data attributes such as decibel levels, traffic density, and green space area, among others. This dataset was fully developed, originated, and exclusively created by Emirhan Bulut at kaggle.com, aceesed on 14 July 2024. It contains essential features and measurements from diverse cities worldwide, emphasizing factors that influence each city's overall happiness score [45]. Preprocessing was a critical initial step, where the raw data underwent rigorous cleaning and normalization to ensure uniformity and accuracy in the subsequent analyses. The pseudocode provided in Algorithm 1 delineates the algorithmic steps involved in this phase, ensuring systematic execution of these tasks.

Algorithm 1 Data Collection and Preprocessing Pipeline

Require: $\mathcal{D}raw$: Raw City Happiness Index Dataset, $\mathcal{F}$: Set of Features $f_1, f_2, \ldots, f_n$, where n denotes the number of features
Ensure: $\mathcal{D}preprocessed$: Preprocessed Dataset

1: Load $\mathcal{D}raw$
2: **for** each feature $f_i \in \mathcal{F}$ **do**
3: 　　Handle missing values using $f_i^{(missing)} \leftarrow \mathcal{M}(f_i)$, where $\mathcal{M}$ represents the chosen imputation strategy
4: 　　Normalize feature f_i to obtain $f_i^{(norm)} \leftarrow \frac{f_i - \mu(f_i)}{\sigma(f_i)}$, where $\mu(f_i)$ and $\sigma(f_i)$ denote the mean and standard deviation of f_i
5: 　　Perform feature engineering to derive $f_i^{(engineered)} \leftarrow \Phi(f_i)$, where Φ represents the transformation or extraction function applied to feature f_i
6: **end for**
7: Store the resulting preprocessed dataset $\mathcal{D}preprocessed$

The success of any machine learning model significantly hinges on the quality of the data used and the effectiveness of the preprocessing techniques applied. This section provides a detailed overview of the dataset utilized in this study, covering its composition, sources, and key features. Additionally, it elaborates on the preprocessing methods applied to prepare the data, including the handling of missing values, feature scaling, and encoding categorical features, which are essential steps to ensure that a model performs effectively.

4.1.1. Dataset Overview

The dataset used in this study encompasses urban-level indicators from multiple cities across various months and years, capturing both environmental and socio-economic factors that influence urban happiness. Specifically, the data include the following features. The *City*, *Month*, and *Year* serve as identifiers for each data record, enabling temporal and geographical analysis of urban happiness. The *Decibel Level* represents the average noise pollution levels measured in decibels, reflecting the noise exposure experienced by city residents. The *Traffic Density* is a categorical variable representing traffic conditions, such as low, medium, or high, which has a direct impact on mobility and quality of life. The *Green Space Area* measures the amount of green space available per capita, in square meters, contributing to the residents' physical and mental well-being. The *Air Quality Index (AQI)* is a numerical value indicating the air quality level, where higher values represent more polluted environments. The target variable in this dataset is the *Happiness Score*, which represents the overall happiness of residents based on surveys and various metrics, scaled from negative to positive values. Additionally, the dataset includes a *Cost of Living Index*, which serves as an indicator of the relative cost required to maintain a certain standard of living in each city, and the *Healthcare Index*, a numerical index reflecting the quality and accessibility of healthcare services available to residents. The dataset consists of 545 rows, each representing a unique city, month, and year combination, thereby providing a comprehensive temporal and geographical overview of urban well-being indicators. The diversity of features allows the hybrid model to capture complex relationships between socio-economic, environmental, and urban infrastructure variables, enabling an in-depth analysis of the factors influencing urban happiness. Detailed information of the dataset is presented in Table 1.

Table 1. Data attributes and their descriptions.

Attribute	Description	Data Type	Range	Example Values
City	Name of the city.	Object	N/A	New York, Los Angeles, Chicago
Month	Month of the year.	Object	N/A	January, February, March
Year	Year of observation.	Integer	2024 (single value)	2024
Decibel_Level	Noise level measured in decibels.	Integer	55–70	70, 65, 60
Traffic_Density	Describes traffic conditions in the city.	Object	High, Medium, Low	High, Medium
Green_Space_Area	Percentage of urban area covered by green spaces.	Integer	30–50	35, 40, 30
Air_Quality_Index	Air quality index (lower is better).	Integer	40–65	40, 50, 60
Happiness_Score	Happiness score on a scale of 0–10.	Float	6.5–7.2	6.5, 6.8, 7.0
Cost_of_Living_Index	Cost of living index (higher means more expensive).	Integer	85–110	100, 90, 85
Healthcare_Index	Index measuring healthcare quality (0–100).	Integer	70–85	80, 75, 70

4.1.2. Data Cleaning and Handling Missing Values

The initial step in the data preparation involved data cleaning to ensure the reliability of the dataset, which included the identification and handling of missing values. Let the dataset be represented by a matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$, where n is the number of instances and m is the number of features. Missing values in features like *Air Quality Index*, *Green Space Area*, and *Healthcare Index* were treated to avoid biased or incomplete model training, which could have resulted in unreliable parameter estimates. For continuous numerical features, such as *Decibel Level*, *Air Quality Index*, and *Cost of Living Index*, missing values were imputed using the arithmetic mean of the observed values, as presented in (33)

$$X_{\text{imputed},j} = \frac{1}{|I_j|} \sum_{i \in I_j} X_{ij} \tag{33}$$

where I_j denotes the set of indices without missing values for feature j, and X_{ij} represents the value of the j-th feature for the i-th instance. This imputation technique preserves the central tendency of the data, ensuring that the statistical properties of the feature are maintained and the impact on variance is minimized.

For categorical features such as *Traffic Density*, missing values were imputed using the mode of the observed values, as presented in (34).

$$X_{\text{imputed},j} = \underset{v \in \{X_{ij}\}_{i \in I_j}}{\arg\max} \ \text{count}(v) \tag{34}$$

where $\text{count}(v)$ represents the frequency of occurrence of category v. This strategy ensured that the categorical distribution remained unbiased, avoiding the introduction of artificial variability.

4.1.3. Feature Scaling

Feature scaling was applied to the numerical features in $\mathbf{X}$ to standardize them to a common scale, which is essential when different features have varying magnitudes and units. Let $\mathbf{X}_{\text{num}}$ represent the subset of numerical features in $\mathbf{X}$. The `StandardScaler` from scikit-learn was used to transform each numerical feature X_j, as presented in (35).

$$X_{\text{scaled},ij} = \frac{X_{ij} - \mu_j}{\sigma_j} \tag{35}$$

where μ_j is the mean of feature j, as presented in (36).

$$\mu_j = \frac{1}{n} \sum_{i=1}^{n} X_{ij} \tag{36}$$

and σ_j is the standard deviation of feature j, as presented in (37).

$$\sigma_j = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_{ij} - \mu_j)^2} \tag{37}$$

This transformation ensures that each feature has a mean of zero and a standard deviation of one, as presented in (38).

$$\mathbb{E}[X_{\text{scaled},j}] = 0, \quad \text{Var}(X_{\text{scaled},j}) = 1 \tag{38}$$

This standardization is critical for the gradient-based optimization algorithms used in neural networks, which are sensitive to the scale of the input features.

4.1.4. Encoding Categorical Variables

Categorical features such as *Traffic Density*, denoted by X_{cat}, were encoded using *one-hot encoding* to transform them into a binary representation suitable for machine learning models. Let X_{cat} contain k unique categories, denoted as $\{c_1, c_2, \ldots, c_k\}$. One-hot encoding was performed by creating k new binary columns $\mathbf{Z} = [z_1, z_2, \ldots, z_k]$, where

$$z_{ij} = \begin{cases} 1, & \text{if instance } i \text{ belongs to category } c_j \\ 0, & \text{otherwise} \end{cases}$$

This encoding ensured that no ordinal relationships were implied among the categories, preventing the model from assuming any unintended ranking or ordering.

The final dataset $\mathbf{X}_{\text{final}}$ was formed by concatenating the scaled numerical features $\mathbf{X}_{\text{scaled}}$ and the encoded categorical features $\mathbf{Z}$ as presented in (39).

$$\mathbf{X}_{\text{final}} = [\mathbf{X}_{\text{scaled}}, \mathbf{Z}] \tag{39}$$

This ensured that both numerical and categorical features were appropriately represented in the feature space for model training.

4.1.5. Splitting the Dataset

The processed dataset $\mathbf{X}_{\text{final}}$ was split into *training* and *testing* sets to evaluate the model's performance. Let $D = (\mathbf{X}_{\text{final}}, \mathbf{y})$ represent the entire dataset, where $\mathbf{y}$ is the target vector (*Happiness Score*). The dataset was partitioned as presented in (40).

$$D_{\text{train}}, D_{\text{test}} = \text{split}(D, \text{ratio} = 0.8 : 0.2) \tag{40}$$

where $D_{\text{train}} = (\mathbf{X}_{\text{train}}, \mathbf{y}_{\text{train}})$ contains 80% of the instances and $D_{\text{test}} = (\mathbf{X}_{\text{test}}, \mathbf{y}_{\text{test}})$ contains 20%. The split was stratified based on the target variable $\mathbf{y}$ to maintain a consistent distribution of *Happiness Score* across both sets, minimizing any potential bias during model evaluation.

4.1.6. Feature Engineering

Feature engineering was performed to improve the model's capacity to learn from complex relationships within the data. *Polynomial features* were generated for specific numerical variables to capture potential interactions between features, which are critical for modeling non-linear relationships. For two numerical features, X_1 and X_2, an interaction term was created, as presented in (41).

$$X_{\text{interaction}} = X_1 \times X_2 \tag{41}$$

This polynomial transformation allowed the model to represent relationships of higher order, providing a richer hypothesis space for learning complex patterns that contribute to urban happiness.

Additionally, *temporal features* such as *Month* and *Year* were transformed into cyclical features to account for periodicity. For a temporal variable *Month*, the transformation was carried out using sine and cosine functions, as presented in (42).

$$\text{Month}_{\sin} = \sin\left(\frac{2\pi \cdot \text{Month}}{12}\right), \quad \text{Month}_{\cos} = \cos\left(\frac{2\pi \cdot \text{Month}}{12}\right) \tag{42}$$

This transformation ensured that the cyclical nature of the data was preserved, thereby allowing the model to understand that the end of the year and the beginning are adjacent.

The final dataset used for modeling consisted of scaled numerical features, one-hot encoded categorical features, polynomial interaction terms, and cyclical temporal features. This comprehensive feature space was designed to enable the *GBM + NN hybrid model* to effectively leverage both ensemble learning and deep learning capabilities for the prediction of urban happiness.

4.2. Model Development and Integration

The core of the predictive analysis involved the development and training of two distinct models, the GBM and NN. As described in Section 3, the integration of these models was a nuanced process where the outputs from the GBM served as inputs to the NN, creating a synergistic model that harnesses the predictive power of both methodologies. Algorithm 2 shows the respective pseudocode sections for the model development and integration.

Algorithm 2 Hybrid Model Development and Integration

Require: $\mathcal{D}train$: Preprocessed Training Dataset, $\mathcal{M}GBM$: Gradient Boosting Machine (GBM) Model, $\mathcal{M}NN$: Neural Network (NN) Model
Ensure: $\mathcal{M}Hybrid$: Integrated GBM-NN Model
1: Train $\mathcal{M}GBM$ on $\mathcal{D}train$, optimizing for $\min_{\mathcal{M}GBM} \mathcal{L}GBM(\mathcal{D}train)$, where $\mathcal{L}GBM$ is the loss function associated with GBM
2: Generate predictions $\hat{y}GBM \leftarrow \mathcal{M}GBM(\mathcal{D}train)$
3: Use $\hat{y}GBM$ as the input features for $\mathcal{M}NN$
4: Train $\mathcal{M}NN$ on $\hat{y}GBM$, optimizing $\min \mathcal{M}NN\mathcal{L}NN(\hat{y}GBM)$, where $\mathcal{L}NN$ is the loss function associated with NN
5: Construct the hybrid model $\mathcal{M}Hybrid \leftarrow \mathcal{F}(\mathcal{M}GBM, \mathcal{M}NN)$, where $\mathcal{F}$ is a function combining the GBM and NN models
6: Return the integrated model $\mathcal{M}Hybrid$

4.3. Evaluation and Interpretation

To comprehensively evaluate the efficacy and reliability of the integrated GBM + NN hybrid model, a robust assessment using *k*-fold cross-validation was employed, as outlined in Algorithm 3. This methodology divided the dataset into *k* disjoint subsets, enabling iterative training and testing, to ensure that every instance contributed to both phases. Such an approach not only validated the model's performance on various subsets but also provided a robust measure of its generalizability to unseen urban settings. The performance metrics derived from this evaluation phase played a critical role in assessing the predictive capabilities and robustness of the model. Four key metrics were utilized: root mean squared error (RMSE), mean absolute error (MAE), coefficient of determination (R^2), and mean absolute percentage error (MAPE). These metrics provided a comprehensive view of the model's predictive accuracy, error magnitude, and explanatory power.

The RMSE, as shown in (43), quantifies the standard deviation of the residuals, representing the average magnitude of prediction errors. This metric is particularly effective in penalizing large errors, making it sensitive to significant deviations between the predicted and actual values.

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{43}$$

Furthermore, the MAE, as presented in (44), measures the average absolute difference between predicted and actual values. Unlike RMSE, it treats all errors equally, providing a straightforward interpretation of prediction accuracy.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{44}$$

Then, the R^2 metric, defined in (45), evaluated the proportion of variance in the target variable explained by the model. A value closer to 1 indicated that the model accounted for most of the variability, reflecting strong predictive power.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \tag{45}$$

Finally, the MAPE, as shown in (46), computes the average percentage difference between predicted and actual values, normalized by the true values. It provides an intuitive measure of prediction accuracy in relative terms.

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \tag{46}$$

Each metric complemented the others, offering a holistic understanding of the model's strengths and limitations. For example, while RMSE penalizes larger errors and highlights significant outliers, MAE provides an unbiased average error magnitude. Meanwhile, R^2 assessed the explanatory power of the model, and MAPE contextualized the errors in percentage terms, enhancing the interpretability for decision-making in urban analytics. In addition, the research culminated in the interpretation and reporting stage, where the results were analyzed to extract meaningful and actionable insights. This analysis focused on understanding the significance of the different predictors and their impact on urban happiness, facilitated by detailed visualizations and comprehensive discussions.

Algorithm 3 Model Evaluation via k-Fold Cross-Validation

Require: $\mathcal{M}Hybrid$: Integrated GBM-NN Model, $\mathcal{D}complete$: Complete Dataset, k: Number of folds

Ensure: $\mathcal{P}$: Performance Metrics (e.g., Accuracy, Precision, Recall, F1-Score)

1: Partition $\mathcal{D}complete$ into k disjoint subsets $\mathcal{D}1, \mathcal{D}2, \ldots, \mathcal{D}k$, where $\mathcal{D}i \cap \mathcal{D}j = \varnothing$ for $i \neq j$ and $\cup i = 1^k \mathcal{D}i = \mathcal{D}complete$
2: **for** each fold $i \in 1, 2, \ldots, k$ **do**
3:     Set $\mathcal{D}test \leftarrow \mathcal{D}i$ and $\mathcal{D}train \leftarrow \mathcal{D}complete \setminus \mathcal{D}i$
4:     Train $\mathcal{M}Hybrid$ on $\mathcal{D}train$ by minimizing the objective function $\min \mathcal{M}Hybrid\mathcal{L}(\mathcal{D}train)$, where $\mathcal{L}$ denotes the model loss function
5:     Test $\mathcal{M}Hybrid$ on $\mathcal{D}test$ to generate predictions $\hat{y}test$
6:     Compute performance metrics $\mathcal{P}i \leftarrow \mathcal{E}(\hat{y}test, y_{test})$, where $\mathcal{E}$ represents the evaluation metric functions and y_{test} denotes the true labels
7: **end for**
8: Compute the average performance $\mathcal{P}avg \leftarrow \frac{1}{k} \sum i = 1^k \mathcal{P}i$
9: Return $\mathcal{P}avg$: Average Performance Metrics

4.4. Statistical Analysis

This section describes the detailed experimental framework used to quantitatively assess the relationship between the urban features and happiness, based on rigorous statistical testing and model interpretability techniques. The goal of these experiments was to determine the individual and joint effects of urban features on the happiness score. These experiments employed cross-validation, hypothesis testing, and regression analysis to derive robust and interpretable results.

4.4.1. Experiment Design and Setup

The dataset $D = (\mathbf{X}, \mathbf{y})$, where $\mathbf{X} \in \mathbb{R}^{n \times m}$ is the feature matrix of urban indicators and $\mathbf{y} \in \mathbb{R}^n$ is the vector of happiness scores, served as the basis for the experiments. The objective was to quantify how the individual features X_j influenced the target variable y. The urban features included indicators like *Air Quality Index (AQI)*, *Traffic Density*, *Green Space Area*, *Healthcare Index*, and *Cost of Living Index*, among others. The experiments were structured to evaluate each feature X_j, or combinations of features, in predicting happiness. The testing procedure involved comparing the predicted happiness scores against the actual values and conducting hypothesis testing to establish the statistical significance of the relationships. Formally, the experiments tested the null hypothesis H_0 (that a feature has no significant effect on happiness, i.e., $\beta_j = 0$) against the alternative hypothesis H_1 (that the feature does have a significant effect, i.e., $\beta_j \neq 0$).

4.4.2. Data Splitting and Cross-Validation

To ensure the robustness of the experiments and prevent overfitting, we used *k-fold cross-validation* with $k = 10$. The dataset was divided into k equally sized subsets or *folds*, denoted $D_1, D_2, \ldots, D_k$. At each iteration, the model was trained on $k - 1$ folds and tested on the remaining folds. This process was repeated k times, with each fold serving as the test set once, thereby ensuring that each instance in the dataset was tested exactly once. In addition, for the hyperparameter tuning, we employed a grid search method to find the most optimal parameter for each model. The overall *cross-validation error E* was calculated as the average error across all folds. For each fold D_i, the error E_i was computed as presented in (47).

$$E_i = \frac{1}{|D_i|} \sum_{j \in D_i} (y_j - \hat{y}_j)^2 \tag{47}$$

where y_j is the actual happiness score for instance j, and $\hat{y}_j$ is the predicted happiness score from the model. The final cross-validation error E was the mean of the errors from all folds, as presented in (48).

$$E = \frac{1}{k} \sum_{i=1}^{k} E_i \tag{48}$$

This approach helped mitigate overfitting by ensuring that the model was evaluated on unseen data in each fold, providing an unbiased estimate of its performance.

4.4.3. Feature Importance and Impact Quantification

The first step in understanding the impact of individual urban features on happiness was to compute *feature importance scores* using the GBM part of the hybrid model. A GBM constructs an ensemble of decision trees, and the feature importance is derived based on how often a feature f_j is used for splitting and the resulting reduction in the loss function. For each feature f_j, the importance score $I(f_j)$ was calculated as (49).

$$I(f_j) = \sum_{t \in T_j} \Delta L(t) \tag{49}$$

where T_j represents the set of decision trees in the ensemble where the feature f_j was used, and $\Delta L(t)$ is the reduction in the loss function $L(y, \hat{y})$ at tree t. The loss function $L(y, \hat{y})$ used in this regression task was the *Mean Squared Error (MSE)*, defined as (50).

$$L(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{50}$$

The feature importance scores provided a preliminary understanding of which features had the most significant impact on happiness.

4.4.4. Pearson Correlation Analysis

To further examine the linear relationships between urban features and happiness, we performed *Pearson correlation analysis*. The Pearson correlation coefficient $\rho(X_j, y)$ was used to measure the linear relationship between each feature X_j and the happiness score y. The Pearson coefficient is defined as (51).

$$\rho(X_j, y) = \frac{\text{Cov}(X_j, y)}{\sigma_{X_j} \sigma_y} \tag{51}$$

where $\text{Cov}(X_j, y)$ represents the covariance between feature X_j and the target variable y, and σ_{X_j} and σ_y are the standard deviations of X_j and y, respectively. The covariance $\text{Cov}(X_j, y)$ was calculated as (52).

$$\text{Cov}(X_j, y) = \frac{1}{n} \sum_{i=1}^{n} (X_{ij} - \bar{X}_j)(y_i - \bar{y}) \tag{52}$$

where $\bar{X}_j$ and $\bar{y}$ represent the mean of the feature X_j and the mean happiness score, respectively. A Pearson correlation coefficient ρ close to 1 or -1 indicates a strong positive or negative linear relationship, respectively, between the feature and happiness.

4.4.5. Hypothesis Testing and Significance Analysis

To establish the statistical significance of the relationship between urban features and happiness, *t*-tests were conducted. The *t*-test was used to compare the means of two groups, such as cities with high air quality versus cities with low air quality, to determine if the difference in happiness scores was statistically significant. The t-statistic for comparing two groups was calculated as (53).

$$t = \frac{\bar{y}_1 - \bar{y}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \tag{53}$$

where $\bar{y}_1$ and $\bar{y}_2$ are the mean happiness scores of the two groups, s_1^2 and s_2^2 are the sample variances, and n_1 and n_2 are the sample sizes for each group. The *degrees of freedom (df)* for the t-test were calculated as (54).

$$df = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{n_1 - 1} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{n_2 - 1}} \tag{54}$$

The resulting *p*-value from the t-test was compared to a significance level $\alpha = 0.05$. If $p < 0.05$, the null hypothesis H_0 (that there was no effect) was rejected, indicating that the feature had a statistically significant effect on happiness. For example, we conducted a t-test comparing happiness scores between cities with high air quality (AQI $\leq$ 50) and

cities with low air quality (AQI > 100). The result showed that improving air quality had a significant positive effect on happiness, with $p < 0.05$.

4.5. Regression Analysis for Marginal Effects

To quantify the magnitude of the effect of each feature, we applied *linear regression analysis*. The linear regression model is given by (55).

$$y_i = \beta_0 + \sum_{j=1}^{m} \beta_j X_{ij} + \epsilon_i \tag{55}$$

where y_i is the happiness score for instance i, X_{ij} is the value of feature X_j for instance i, and β_j is the regression coefficient representing the marginal effect of X_j on y. The error term ϵ_i represents the residual, or the difference between the predicted and actual happiness score. The regression coefficients β_j were estimated by minimizing the *Residual Sum of Squares (RSS)* as presented as (56).

$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{56}$$

where $\hat{y}_i$ represents the predicted happiness score for instance i. The statistical significance of each coefficient β_j was assessed using *t*-tests on the regression coefficients, with corresponding *p*-values used to determine if the effect of each feature was significant. For example, a 10% improvement in air quality led to an estimated 5% increase in happiness, with a *p*-value < 0.01, confirming the significance of the result.

5. Result and Discussion

The performance of the various machine learning models for the prediction task was evaluated using 10-fold cross-validation, and the results are summarized in Table 2. Key performance metrics included the average root mean square error (RMSE), average mean absolute error (MAE), average coefficient of determination (R^2), and average mean absolute percentage error (MAPE). First, the GBM + NN hybrid model achieved the best overall performance across all metrics, with an RMSE of 0.3332, MAE of 0.2633, R^2 of 0.9673, and MAPE of 7.0082%. The low RMSE and MAE values indicated high predictive accuracy, while the R^2 value showed that 96.73% of the variance in the target variable was explained by the model. The low MAPE further highlighted the model's robustness in minimizing percentage errors. This superior performance can be attributed to the hybrid nature of the model, which combines the structured data handling capabilities of GBM with the non-linear feature extraction capabilities of neural networks. Furthermore, tree-based models such as the random forest, gradient boosting machine (GBM), and CatBoost performed competitively, with the random forest achieving an RMSE of 0.4063, MAE of 0.3173, R^2 of 0.9524, and MAPE of 11.86%. CatBoost achieved slightly better RMSE and MAE values compared to the GBM but lagged behind GBM + NN and random forest in overall performance. With an RMSE of 0.8189 and R^2 of 0.8120, the GBM demonstrated good predictive capability but was surpassed by GBM + NN and random forest. On the other hand, CatBoost achieved the lowest RMSE (0.3486) among the individual tree-based models, reflecting a strong predictive accuracy. However, its MAPE (8.4328%) was slightly higher than GBM + NN, indicating room for improvement in capturing percentage-based errors.

Table 2. Ten-Fold Cross-Validation Results for Various Models

Model	Average RMSE	Average MAE	Average R^2	Average MAPE (%)
Dense Neural Network	0.5837	0.4342	0.8949	69.6198
LSTM + CNN	1.2188	0.9900	0.3955	67.3178
CNN	0.4923	0.3673	0.9227	69.4898
DeepGBM	0.5626	0.4658	0.9028	67.9763
Ensemble Model	1.5114	1.2648	0.3398	48.8259
GRU	0.4931	0.3783	0.9226	69.2551
LSTM	1.0239	0.8424	0.5992	67.9094
Autoencoder + Regression	0.6566	0.4993	0.8679	68.5552
ResNet	0.6677	0.5239	0.8655	69.5246
MLP	0.6031	0.4653	0.8894	69.6376
GBM	0.8189	0.6787	0.8120	25.8416
Linear Regression	0.5485	0.4280	0.9136	10.9827
TabNet	5.6100	5.1469	−8.5989	84.3540
GBM + NN	**0.3332**	**0.2633**	**0.9673**	**7.0082**
CatBoost Regressor	1.1114	0.9088	0.6519	36.7200
Random Forest Regressor	0.4063	0.3173	0.9524	11.8600

Among neural network models, the dense neural network and convolutional neural network (CNN) showed a competitive performance. The CNN achieved an RMSE of 0.4923, MAE of 0.3673, and R^2 of 0.9227, outperforming many other neural network models. The dense neural network exhibited an RMSE of 0.5837 and R^2 of 0.8949, suggesting a good overall performance, but not as strong as the CNN. The other neural network architectures like GRU (RMSE: 0.4931, R^2: 0.9226) and ResNet (RMSE: 0.6677, R^2: 0.8655) showed moderate results, indicating their potential for handling temporal and spatial data, albeit less effectively for this task. The standalone ensemble model performed poorly compared to its counterparts, with an RMSE of 1.5114, MAE of 1.2648, and R^2 of only 0.3398. The high MAPE (48.8259%) suggests that this approach struggled to generalize effectively on the dataset. Furthermore, the inclusion of temporal structures in models such as LSTM and LSTM + CNN did not yield favorable results. LSTM had an RMSE of 1.0239 and R^2 of 0.5992, indicating limited effectiveness in capturing patterns in this dataset. LSTM + CNN performed worse, with an RMSE of 1.2188 and R^2 of 0.3955, suggesting that the combination of temporal and spatial features did not synergize well for this task.

Next, traditional regression approaches, such as linear regression, showed respectable results, with an RMSE of 0.5485, MAE of 0.4280, R^2 of 0.9136, and MAPE of 10.9827%. This indicates that linear models can capture significant patterns in data but fall short compared to more advanced methods. TabNet showed the poorest performance across all metrics, with an RMSE of 5.6100 and a negative R^2 value (−8.5989), indicating that the model failed to fit the data effectively. Autoencoder + Regression performed moderately, with an RMSE of 0.6566 and R^2 of 0.8679, but did not outperform the tree-based or hybrid models. The results demonstrate the significant advantage of hybrid models like GBM + NN, which combine the strengths of traditional tree-based methods and deep learning architectures. Models like the random forest and CatBoost consistently delivered a strong performance, highlighting their effectiveness in handling structured, tabular data. While the CNN and dense neural networks showed strong performance, architectures like LSTM and ResNet were less effective, emphasizing the importance of choosing the right neural network for specific tasks. The poor performance of TabNet suggests that it may not be well-suited for this dataset, possibly due to overfitting or difficulties in feature representation. The GBM + NN hybrid model was the most effective approach for this task,

achieving the best performance across all metrics. Future research could explore optimizing hybrid architectures further and investigating feature engineering techniques to enhance model performance. Additionally, understanding the limitations of the underperforming models like TabNet could provide insights into dataset-specific challenges. Beside the comparison of the machine learning and deep learning models, we also have the results of the statistical experiments, Table 3 demonstrates that several key urban features had a statistically significant and substantial impact on happiness. A 10% improvement in air quality led to a 5% increase in happiness, with a p-value of 0.01, confirming its significance. Reducing traffic density from high to medium resulted in a 4% increase in happiness, while increasing green space by 1 square meter per person was associated with a 3% increase in happiness, both with p-values below 0.05. These results were validated through cross-validation and hypothesis testing, providing robust evidence for the relationships between urban features and happiness.

Table 3. Impact of Key Urban Features on Happiness

Urban Feature	Change in Feature	Change in Happiness	p-Value
Air Quality	10% improvement in AQI	5% increase	0.01
Traffic Density	High to Medium	4% increase	0.03
Green Space	+1 m^2 per person	3% increase	0.04
Cost of Living Index	−5% decrease	2.5% increase	0.02
Healthcare Index	+10% improvement	3.5% increase	0.01

6. Conclusions

This study proposed a novel hybrid approach combining GBM and NN models for the prediction of urban happiness. By leveraging the capabilities of ensemble learning in GBMs and the deep feature extraction in neural networks, the GBM + NN hybrid model achieved significant improvements in predictive accuracy compared to other traditional machine learning and deep learning models. The experimental results demonstrated that the hybrid model outperformed all other models tested, achieving the lowest RMSE of 0.3383. The effectiveness of the hybrid model can be attributed to its ability to effectively capture complex feature interactions and refine predictions through a two-stage learning process. This approach not only improved the accuracy of predictions but also provided valuable insights into the key factors influencing urban happiness, such as air quality, traffic density, green space availability, healthcare quality, and cost of living. These insights can serve as a valuable resource for urban planners and policymakers in developing evidence-based interventions aimed at enhancing the quality of life in cities.

The comparative analysis of the GBM + NN hybrid model against models such as DeepGBM, CNN, ResNet, and TabNet further highlighted the advantages of integrating ensemble learning with deep learning techniques. Models like CNN and DeepGBM performed reasonably well, but the absence of an integrated learning structure limited their predictive capabilities relative to the hybrid model. Traditional models like linear regression and random forest failed to capture the non-linear relationships between urban features adequately, leading to higher prediction errors. The findings of this study emphasize the importance of adopting hybrid models for complex prediction tasks, where a combination of structured feature handling and deep representation learning is required. The GBM + NN hybrid model presents a new benchmark in urban happiness prediction, showcasing a promising direction for future research that involves the integration of different machine learning paradigms to enhance model performance. Future research could explore

the extension of this hybrid approach by incorporating additional contextual features, such as real-time social media data, mobility patterns, and climate information, to further improve the model's predictive capabilities. Additionally, the interpretability of the hybrid model could be enhanced by applying feature importance techniques and explainable AI methods to provide a more transparent understanding of the impact of each predictor on urban happiness.

Author Contributions: Conceptualization, G.A. and A.L.; methodology, G.A.; software, G.A.; validation, G.A. and A.L.; formal analysis, G.A.; investigation, G.A.; resources, G.A.; data curation, G.A.; writing—original draft preparation, G.A.; writing—review and editing, A.L.; visualization, G.A.; supervision, A.L.; project administration, A.L.; funding acquisition, A.L. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: The data supporting the reported results can be accessed from https://www.kaggle.com/datasets/emirhanai/city-happiness-index-2024 (accessed on 14 July 2024). In cases where no new data were created or where data is unavailable due to privacy or ethical restrictions, this statement must be updated.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Mouratidis, K. Urban planning and quality of life: A review of pathways linking the built environment to subjective well-being. *Cities* **2021**, *115*, 103229. [CrossRef]
2. Sheikh, W.T.; van Ameijde, J. Promoting livability through urban planning: A comprehensive framework based on the "theory of human needs". *Cities* **2022**, *131*, 103972. [CrossRef]
3. Mouratidis, K. COVID-19 and the compact city: Implications for well-being and sustainable urban planning. *Sci. Total Environ.* **2022**, *811*, 152332. [CrossRef] [PubMed]
4. Patino, J.E.; Martinez, L.; Valencia, I.; Duque, J.C. Happiness, life satisfaction, and the greenness of urban surroundings. *Landsc. Urban Plan.* **2023**, *237*, 104811. [CrossRef]
5. Krekel, C.; MacKerron, G. How environmental quality affects our happiness. In *World Happiness Report*; Sustainable Development Solutions Network: New York, NY, USA, 2020; pp. 95–112.
6. Addas, A. Influence of urban green spaces on quality of life and health with smart city design. *Land* **2023**, *12*, 960. [CrossRef]
7. Wójcik, P.; Andruszek, K. Predicting intra-urban well-being from space with nonlinear machine learning. *Reg. Sci. Policy Pract.* **2022**, *14*, 891–914. [CrossRef]
8. Liu, G.; Ma, J.; Chai, Y. Nonlinear relationship between microenvironmental exposure and travel satisfaction explored with machine learning. *Transp. Res. Part D Transp. Environ.* **2024**, *128*, 104104. [CrossRef]
9. Ma, J.; Dong, G. Periodicity and variability in daily activity satisfaction: Toward a space-time modeling of subjective well-being. *Ann. Am. Assoc. Geogr.* **2023**, *113*, 1918–1938. [CrossRef]
10. Ohanyan, H.; Portengen, L.; Huss, A.; Traini, E.; Beulens, J.W.; Hoek, G.; Lakerveld, J.; Vermeulen, R. Machine learning approaches to characterize the obesogenic urban exposome. *Environ. Int.* **2022**, *158*, 107015. [CrossRef] [PubMed]
11. Kumar, V.; Kedam, N.; Sharma, K.V.; Khedher, K.M.; Alluqmani, A.E. A comparison of machine learning models for predicting rainfall in urban metropolitan cities. *Sustainability* **2023**, *15*, 13724. [CrossRef]
12. Costa, V.G.; Pedreira, C.E. Recent advances in decision trees: An updated survey. *Artif. Intell. Rev.* **2023**, *56*, 4765–4800. [CrossRef]
13. Linka, K.; Hillgärtner, M.; Abdolazizi, K.P.; Aydin, R.C.; Itskov, M.; Cyron, C.J. Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning. *J. Comput. Phys.* **2021**, *429*, 110010. [CrossRef]
14. Fitz, S.; Romero, P. Neural networks and deep learning: A paradigm shift in information processing, machine learning, and artificial intelligence. In *The Palgrave Handbook of Technological Finance*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 589–654.
15. Cheung, E.Y.; Wu, R.W.; Li, A.S.; Chu, E.S. AI deployment on GBM diagnosis: A novel approach to analyze histopathological images using image feature-based analysis. *Cancers* **2023**, *15*, 5063. [CrossRef]

16. Liu, M.; Chen, H.; Wei, D.; Wu, Y.; Li, C. Nonlinear relationship between urban form and street-level PM2.5 and CO based on mobile measurements and gradient boosting decision tree models. *Build. Environ.* **2021**, *205*, 108265. [CrossRef]

17. Cerono, G.; Melaiu, O.; Chicco, D. Clinical feature ranking based on ensemble machine learning reveals top survival factors for glioblastoma multiforme. *J. Healthc. Inform. Res.* **2024**, *8*, 1–18. [CrossRef] [PubMed]

18. Abdolrasol, M.G.; Hussain, S.S.; Ustun, T.S.; Sarker, M.R.; Hannan, M.A.; Mohamed, R.; Ali, J.A.; Mekhilef, S.; Milad, A. Artificial neural networks based optimization techniques: A review. *Electronics* **2021**, *10*, 2689. [CrossRef]

19. Hollmann, N.; Müller, S.; Eggensperger, K.; Hutter, F. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv* **2022**, arXiv:2207.01848.

20. Gallagher, K.R. *Bridging the Gap Between Science and Practice: Examining if Conceptual Models can be Effective as Tools to Guide the Planning and Valuation of Multi-Use Urban Trails*; The University of Toledo: Toledo, OH, USA, 2021.

21. Khreis, H. Traffic, air pollution, and health. In *Advances in Transportation and Health*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 59–104.

22. Samal, S.R.; Mohanty, M.; Santhakumar, S.M. Adverse effect of congestion on economy, health and environment under mixed traffic scenario. *Transp. Dev. Econ.* **2021**, *7*, 15. [CrossRef]

23. Rahman, M.M.; Najaf, P.; Fields, M.G.; Thill, J.C. Traffic congestion and its urban scale factors: Empirical evidence from American urban areas. *Int. J. Sustain. Transp.* **2022**, *16*, 406–421. [CrossRef]

24. Castelli, C.; d'Hombres, B.; Dominicis, L.d.; Dijkstra, L.; Montalto, V.; Pontarollo, N. What makes cities happy? Factors contributing to life satisfaction in European cities. *Eur. Urban Reg. Stud.* **2023**, *30*, 319–342. [CrossRef]

25. Tan, M.J.; Guan, C. Are people happier in locations of high property value? Spatial temporal analytics of activity frequency, public sentiment and housing price using twitter data. *Appl. Geogr.* **2021**, *132*, 102474. [CrossRef]

26. Das, K.V.; Jones-Harrell, C.; Fan, Y.; Ramaswami, A.; Orlove, B.; Botchwey, N. Understanding subjective well-being: Perspectives from psychology and public health. *Public Health Rev.* **2020**, *41*, 25. [CrossRef] [PubMed]

27. Koumetio Tekouabou, S.C.; Diop, E.B.; Azmi, R.; Chenal, J. Artificial intelligence based methods for smart and sustainable urban planning: A systematic survey. *Arch. Comput. Methods Eng.* **2023**, *30*, 1421–1438. [CrossRef]

28. Quak, D.; Luetz, J.M. Human happiness: Conceptual and practical perspectives. In *No Poverty*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 459–475.

29. Bettencourt, L.M. *Introduction to Urban Science: Evidence and Theory of Cities as Complex Systems*; MIT Press: Cambridge, MA, USA, 2021.

30. Saha, K. Computational and Causal Approaches on Social Media and Multimodal Sensing Data: Examining Wellbeing in Situated Contexts. Ph.D. Dissertation, Georgia Institute of Technology, Atlanta, GA, USA, 2021.

31. Iacus, S.M.; Porro, G. *Subjective Well-Being and Social Media*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2021.

32. Saha, K.; De Choudhury, M. Examining Well-Being in Situated Contexts with Computational Modeling of Social Media Data. In *Mobile Sensing in Psychology: Methods and Applications*;The Guilford Press: New York, NY, USA, 2023 ; p. 215.

33. Zareba, M.; Cogiel, S.; Danek, T.; Weglinska, E. Machine Learning Techniques for Spatio-Temporal Air Pollution Prediction to Drive Sustainable Urban Development in the Era of Energy and Data Transformation. *Energies* **2024**, *17*, 2738. [CrossRef]

34. Jun, M.J. A comparison of a gradient boosting decision tree, random forests, and artificial neural networks to model urban land use changes: The case of the Seoul metropolitan area. *Int. J. Geogr. Inf. Sci.* **2021**, *35*, 2149–2167. [CrossRef]

35. Mondal, S.; Ghosh, S.; Nag, A. Brain stroke prediction model based on boosting and stacking ensemble approach. *Int. J. Inf. Technol.* **2024**, *16*, 437–446. [CrossRef]

36. Luo, J.; Xu, S. NCART: Neural Classification and Regression Tree for tabular data. *Pattern Recognit.* **2024**, *154*, 110578. [CrossRef]

37. Rithani, M.; Kumar, R.P.; Doss, S. A review on big data based on deep neural network approaches. *Artif. Intell. Rev.* **2023**, *56*, 14765–14801. [CrossRef]

38. Khan, A.; Fouda, M.M.; Do, D.T.; Almaleh, A.; Rahman, A.U. Short-term traffic prediction using deep learning long short-term memory: Taxonomy, applications, challenges, and future trends. *IEEE Access* **2023**, *11*, 94371–94391. [CrossRef]

39. Zhang, Y.; Zong, R.; Shang, L.; Kou, Z.; Zeng, H.; Wang, D. Crowdoptim: A crowd-driven neural network hyperparameter optimization approach to ai-based smart urban sensing. *Proc. ACM Hum.-Comput. Interact.* **2022**, *6*, 1–27. [CrossRef]

40. Liu, X.; Hu, Q.; Li, J.; Li, W.; Liu, T.; Xin, M.; Jin, Q. Decoupling representation contrastive learning for carbon emission prediction and analysis based on time series. *Appl. Energy* **2024**, *367*, 123368. [CrossRef]

41. Dixon, J.; Akinniyi, O.; Abdelhamid, A.; Saleh, G.A.; Rahman, M.M.; Khalifa, F. A hybrid learning-architecture for improved brain tumor recognition. *Algorithms* **2024**, *17*, 221. [CrossRef]

42. Xie, P.; Li, T.; Liu, J.; Du, S.; Yang, X.; Zhang, J. Urban flow prediction from spatiotemporal data using machine learning: A survey. *Inf. Fusion* **2020**, *59*, 1–12. [CrossRef]

43. Wei, P.; Hao, S.; Shi, Y.; Anand, A.; Wang, Y.; Chu, M.; Ning, Z. Combining Google traffic map with deep learning model to predict street-level traffic-related air pollutants in a complex urban environment. *Environ. Int.* **2024**, *191*, 108992. [CrossRef] [PubMed]

44. Kim, C.; Park, T. Predicting determinants of lifelong learning intention using gradient boosting machine (GBM) with grid search. *Sustainability* **2022**, *14*, 5256. [CrossRef]

45. Bulut, E. City Happiness Index 2024. Available online: https://www.kaggle.com/datasets/emirhanai/city-happiness-index-2024 (accessed on 14 July 2024).