Data information prediction based on deep fusion GRU-Stacking

1st Quanli Pei
College of Computer and Information Engineering
Hubei Normal University
Huangshi, China
leon_peiquanli@163.com

2nd Yulong Chen
College of Computer and Information Engineering
Hubei Normal University
Huangshi, China
ylchen0424@stu.hbnu.edu.cn

3rd Yu Liu
College of Computer and Information Engineering
Hubei Normal University
Huangshi, China
liuyuuuu1110@gmail.com

4th Teng Li
College of Computer and Information Engineering
Hubei Normal University
Huangshi, China
liteng9902@gmail.com

5th Wei Zhang
College of Computer and Information Engineering
Hubei Normal University
Huangshi, China
wei947008@gmail.com

6th Wenrui Xiong
College of Computer and Information Engineering
Hubei Normal University
Huangshi, China
xwrrrrrr20033333@gmail.com

7th Xinpin Jiang
College of Computer and Information Engineering
Hubei Normal University
Huangshi, China
jiangxinping016@gmail.com

8st Min Pan
College of Computer and Information Engineering
Hubei Normal University
Huangshi, China
panminiii@mails.ccnu.edu.cn

Abstract—**Machine learning and deep learning are currently widely used to predict data information. The purpose of this paper is to present our contribution to this field by discussing a common event data prediction problem-bicycle sharing demand prediction. The Bicycle Sharing Project has significantly reduced resource consumption and air pollution, but it suffers from the inability of the responsible authorities to predict the demand for bicycles at each station based on spatial and temporal data from the stations, which could result in the use of unnecessary human resources to dynamically balance the number of bicycles at each station. In our work, we propose a GRU-Stacking based model to predict the demand for shared bicycles to solve the above problems. The first level of the prediction model is built using the stacking ensemble learning strategy, the second level is built using the two-layer GRU model, and finally the results of the two levels are fused by weighted average fusion to deeply integrate the two models in order to improve the prediction accuracy of the model. The results demonstrated that our model outperforms popular prediction models like XGBoost, LightGBM, LCE and Bi-LSTM on three evaluation metrics, R^2, RMSLE and MAPE. This shows that our model is well suited to the task of prediction the demand for shared bicycles and can be used to solve other problems involving data information prediction.**

Keywords—Data information prediction, Bike-Sharing demand, Stacking, GRU, Deep fusion

I. INTRODUCTION

To present, the usage of motor vehicles has resulted in substantial air pollution due to the release of greenhouse gases from the combustion of fossil fuels. Statistics show that the usage of motor vehicles is responsible for around 40% of the greenhouse gas emissions in the European region [1,2]. Action is required to change people's travel habits from using motorized cars to more ecologically friendly means, which is why bike-sharing projects were established. Bike sharing projects contribute to a cleaner, more cost-effective, and healthier urban transportation networks by lowering cities' air pollution and resource consumption [3, 4].

Bike-sharing projects have been widely adopted in many nations and areas, but this has led to a variety of issues. The relevant authorities, for instance, are unable to dynamically estimate the demand for bicycles at each station based on spatial and temporal characteristics, making it impossible to foresee the idle and saturated states of bicycle stations at a given moment [5,6]. To address the issues with the aforementioned bicycle sharing projects, the purpose of this article is to employ sophisticated models to forecast the future demand for bicycles at each station.

Stacking is a hierarchical model ensemble learning strategy in which one or more base learners may make up each layer. When compared to employing a single machine learning model, stacking models often generalize significantly better and can produce more accurate predictions [7]. With the help of their innovative "gating" mechanisms, deep learning models LSTM and GRU have been developed to address Spatio-Temporal issues associated with short-term memory. These models send pertinent input through lengthy chains of sequences for prediction[8,9].

Both approaches, while strong in specific cases, are somewhat limited to sizes and types of data. The Stacking method successfully combines the advantages of multiple conventional machine learning models and excels at prediction tasks, but it struggles to capture the dynamism of spatiotemporal properties in the dataset for bike sharing. Although they need a lot of detailed data, deep learning models like LSTM and GRU can forecast bike-sharing demand accurately in the short term.

We aim to apply a fusion of the two models to the problem of forecasting demand for shared bicycles while taking into account the shortcomings of each of the aforementioned two models. By capturing the dynamic changes in the Spatio-temporal properties of the bike-sharing dataset, we anticipate that this concept can, on the one hand, address the two issues outlined above. On the other hand, it can enhance the model's overall generalization capability with little data and produce better prediction outcomes. We specifically build a GRU-StackiTong-based bicycle sharing demand forecasting model, integrating Random Forest Regression, Bagging Regression, XGBoost, and LightGBM first, then using the Stacking integrated model as the forecasting model's first level model. The second-level model of the prediction model, which together makes up our prediction model, is then based on the

978-1-6654-9402-1/22/$31.00 ©2022 IEEE
DOI 10.1109/WI-IAT55865.2022.00140

two-level GRU model. To calculate the weights using the entropy weighting approach and produce the final model results, the results of the first two models mentioned above are used.

Our suggested model significantly outperforms models based on LEC, two-layer LSTM, Bi-GRU, and Bi-LSTM on three individual metric scores, R^2, RMSLE and MAPE, on the London Bike Sharing dataset provided by the Kaggle competition platform, demonstrating the model's superiority.

The article is set up as follows: Our research on three predicting components of bike-sharing demand, the Stacking integration approach, and the use of deep learning for spatiotemporal issues are presented in Section II. Section III introduces our model, which was developed using feature analysis, data processing, and data description, and it also outlines the model's assessment metrics. The stacking integration procedure and the outcomes of the multi-model comparison are presented and discussed in Section IV. Section V concludes with a discussion of the created prediction model.

II. Related Work

The work on bike-sharing demand prediction and associated issues is directly connected to ours, therefore before we explain our model, we first address their works. Next, we discuss the Stacking integration strategy. Finally, we demonstrate how deep learning is used to spatiotemporal issues.

A. Demand for bicycle sharing prediction

There are two primary areas of current bicycle sharing demand prediction research. The first part is researching the variables that affect the demand for shared bicycles. According to Campbell et alinvestigation .'s of a shared bicycle project in Beijing, the main factors influencing the demand for shared bicycles were distance, temperature, precipitation, and air quality [10]; Ezgi Eren et al. hypothesize that weather conditions, time of day factors, and socio-demographics may have an impact on the demand for shared bicycles [11]; Climate elements including temperature, wind, and precipitation were named by Matton et al. as the primary determinants of the demand for shared bicycles [12].

The development of models to predict the demand for shared bicycles is another component. Neofytos et al. employed machine learning models including Boosting, Gradient, and XGBoost for comparison to predict bicycle sharing demand, and they discovered that the algorithms outperformed their human counterparts[13]. They only employed one machine learning model, though, and stopped working on the integration of these models; To predict the demand for bicycle sharing data in Suzhou, China, Bo Wang et al. employed deep learning models including RNN, LSTM, and GRU, respectively. They discovered that these time-related deep learning models performed well on bicycle sharing demand prediction [5]; Common machine learning methods and deep learning models were employed by Seunghan et al. to predict the demand for shared bikes, and they discovered that the deep learning algorithm was superior [6]. However, the models needed a lot of data to provide accurate predictions. The study described above demonstrates that both machine learning and deep learning models have advantages and room for improvement: (1) Machine learning models often need less data than deep learning models, and they can keep become more efficient by using the ensemble learning strategy we discuss in B. (2) For the job of predicting bike-sharing demand, time-series-related deep learning models are able to outperform machine learning methods and are ideal for tasks incorporating Spatio-temporal information, as we highlight in C.

B. Stacking ensemble learning strategy for Prediction

In a nutshell, the stacking ensemble learning strategy involves stacking models by incorporating new characteristics into existing models using the output results of a number of base models. Consider a two-layer stacking model. The input to the first layer of the base model is the original data. The output of the first layer of the base model is then added to the original data features as input to the second layer of the metamodel.

To enhance the prediction classification of software flaws, Sweta Mehta et al. employed machine learning techniques such as ANN, decision time, KNN, and SVM with Stacking integration [14]. The aforementioned methods were selected because they can categorize software modules in software to predict the development of cardiovascular illness, Jimin Liu et al. employed machine learning algorithms such as SVM, KNN, LR, and RF as base learners and LR as a meta-learner, respectively [15]. The best base learner was chosen by comparing the experimental findings. It is clear that in the model integration through the stacking procedure, the problem is more often than not taken into account while choosing the base model, as opposed to the model's merits. The capacity of the models integrated by stacking to generalize to a specific problem is typically higher than that of individual machine learning, and the filtering of base and metamodels aids in preventing the randomness of integration. Additionally, because machine learning models require less data than deep learning models, we decide to integrate machine learning algorithms using the stacking ensemble learning strategy.

C. Deep learning for Spatio-Temporal problems

Senzhang Wang et al. categorized Spatio-temporal data as event data, trajectory data, point reference data, raster data, and video data [16]. Spatio-temporal issues frequently contain significant amounts of Spatio-temporal data. Deep learning models do well with this type of data. Traditional crowd counting methods cannot take into account the temporal connection between subsequent video frames, but the ST-CNN model developed by Yunqi Miao et al. can feature extract from video data by first abstracting temporal data into spatial data of different dimensions [17]; The ST-GCN model was created by Sijie Yan et al. and is highly generalizable [18]. It can automatically learn spatial and temporal patterns from skeletal data; Farah Shahid et al. demonstrated experimentally that deep learning models like Bi-LSTM may be utilized for illness prediction by using event data to forecast the number of upcoming 2018 coronavirus disease infections using ARIMA, SVR, LSTM, and Bi-LSTM models [19]; To increase the predictability of truck traffic flow, Shengyou Wang et al. employed both LSTM and GRU, two God will network approaches. The experimental findings revealed that these two deep learning models outperformed the machine learning models SRV and Arima [20].

In summary, we chose deep learning models like Bi-LSTM and GRU for the creation of prediction models since we think they can perform well on Spatio-temporal issues incorporating event data.

III. METHODOLOGY

This paper uses the London Bike Sharing Dataset provided by the Kaggle competition platform as a research case, which is a dataset consisting of historical data combined with weather data from the bike-sharing system in the city of London, UK, and collects data from stations with a demand of over 500 bike-sharing stations in the city of London. This section demonstrates three key points: first, the description, correlation analysis, and processing of the London Bike Sharing Dataset's data characteristics; second, the fundamentals of the two-layer GRU model and our application strategy; and third, the proposed GRU-Stacking based bicycle sharing demand forecasting model.

A. Data Description and Analysis

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

The two major categories of time and environment may be used to categorize the data features of the London Bike Sharing Dataset. The environment includes ambient temperature, body temperature, air humidity, wind speed, and weather. The time includes the date (year, month, day, hour, and minute), season (0: spring, 1: summer, 2: autumn, and 3: winter), weekday (0: non-working day, 1: working day), and season (0: non-holiday, 1: holiday) (1: clear, 2: scattered clouds, a 3: broken clouds, 4: cloudy, 7: light rain, 10: thunderstorms, 26: snowfall, 94: freezing fog).

The dataset contains 17,414 data and was timed starting at 00:00 on January 4, 2015, and was updated hourly until 23:00 on January 3, 2017, recording the time, weather, and quantity of cyclists at stations. For this study, the first 70% of all data were used as the training set for the model training and the last 30% as the test set for the model testing. Additionally, we examined how time and environment affect the demand for shared bicycles.

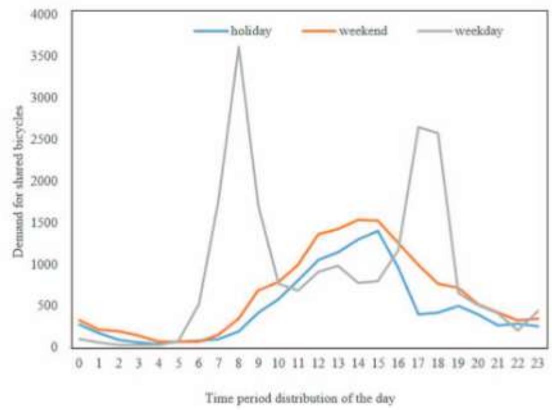

Fig. 1. Trend in the time series of demand for shared bicycles

We used information on the demand for shared bikes in London for a whole year in 2015 in time-series variance analysis. On weekdays, the morning peak (6–10 AM) and evening peak (4–8 PM) had a much greater demand for bikes than other times of the day, as shown in Fig. 1. On weekends and holidays, the temporal pattern in demand for bikes was more or less the same and focused around midday (11-14 AM). In general, weekdays have a higher demand for bicycle sharing than weekends. In conclusion, the demand for bicycle sharing is significantly influenced by the availability of the service throughout the weekdays and the various hours of the day.

We conducted a correlation study utilizing the demand data for the London bike-share program from 2015 to 2017 along with three environmental factors. As seen in Fig. 2, we discovered that demand for bike sharing was connected with all three environmental factors, with wind speed showing the strongest association and air humidity and weather showing the weakest correlation. With a correlation value of -0.46, the demand for bicycle sharing was shown to be most correlated with air humidity.

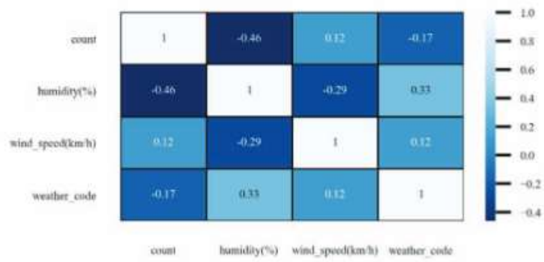

Fig. 2. Demand for shared bikes and environmental conditions are correlated on a heat map

B. Data Processing

Before splitting the dataset, we analyze the temporal category characteristics to take into account their effect on the demand for shared bicycles. The year, month, day, and hour were specifically taken from the date information and added to the dataset as new features before the original date information was deleted. Since the aforementioned data analysis process showed that wind speed and bicycle sharing demand are positively correlated, which is inconsistent with actual life circumstances, we checked the dataset and discovered that wind speed contains a large number of zero values, which may affect the model's prediction outcomes. To fill the original zero values with projected values, we employ a random forest model that adequately matches the data to create zero-value predictions. All that is necessary for data supplied into the Stacking integrated learning model is the aforementioned temporal processing and zero padding. In addition to the processing of the temporal features and the zero-value padding mentioned above, we also utilize normalization to standardize the order of magnitude before the various characteristics in the data are supplied into the deep learning model.

C. Stacking ensemble learning strategy

The stacking ensemble learning strategy mixes many machine learning model types through some sort of fusion, utilizing the benefits of various machine learning models to analyze data from various angles. As a result, the first layer's base learner must choose a variety of models in addition to high-performing ones. To repair the faults of the base learner in the first layer and get the best prediction, the meta-learner in the second layer should choose models with strong generalization potential. A bagging model uses the bootstrap technique to produce aggregated predictors by aggregating several copies of the predictors, which has the benefit of enhancing model accuracy and stability [21]; The Random

Forest (RF) model, which can successfully avoid overfitting and is immune to interference and parallel processing, is a model integrated by bagging [22]; Extreme Gradient Boosting (XGBoost) integrated by boosting with a regularization term to prevent overfitting, with high computational efficiency and multi-threaded operation[23]; The LightGBM model is a Histogram-based decision number algorithm that allows the information gain from large gradient samples to be amplified while retaining large gradient samples[24].

In summary, we consider the combination of Bagging, RF, XGBoost, and LightGBM in different combinations as the base learner of the Stacking integration method, and each of the four models as a meta-learning model, using a five-fold cross-validation approach for Stacking integration, and the integrated model is called the first-level model of the overall prediction model. We created trials to individually forecast the various combinations and assessed the prediction outcomes (the evaluation metrics we used were R^2 and RMSLE in Section III.F). This allowed us to choose the optimal base learner and meta-learner, the model framework is shown in Fig. 3.

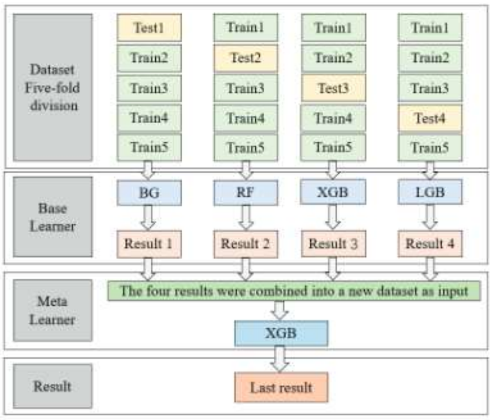

Fig. 3. The model framework obtained after Stacking ensemble learning strategy

D. Two-layer GRU model

The Gated Recurrent Unit (GRU) is a variant of recurrent neural networks that Cho et al. first proposed in 2014 and used in sequence studies [9]. It shares the same gating unit as the LSTM to minimize information loss during the propagation of long sequences (long-term) through the gating unit, but it lacks a recurrent memory unit, making it less complex than the LSTM. In a different research, Cho demonstrated that GRU outperforms LSTM with a sufficient dataset and is quicker than LSTM [25]. The mathematical expressions can be denoted as

$$z_t=\sigma(W_z[h_{t-1},x_t]) \tag{1}$$

$$r_t=\sigma(W_r[h_{t-1},x_t]) \tag{2}$$

$$\widetilde{h}_t=\tanh(W[r_t\odot h_{t-1},x_t]) \tag{3}$$

$$h_t=(1-z_t)\odot h_{t-1}+z_t\odot\widetilde{h}_t \tag{4}$$

Inputs x_t and h_{t-1} at time t are passed through the activation function with the weight matrix to obtain the reset gate z_t and update gate r_t respectively. In a similar way, we obtain $\widetilde{h}_t$. We then obtain the hidden state h_t.

In this paper, we have used the two-layer GRU model as a secondary model for the overall prediction model and we can see in Fig. 4 how the two-layer GRU model works when we input x_t.

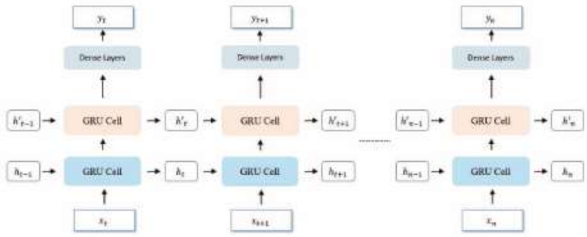

Fig. 4. Two-tier GRU model framework (the Second level of the overall predictive model)

E. GRU-Stacking based bicycle sharing demand prediction model

The model obtained by the Stacking integration method after filtering the base model and meta-model is used as the primary model of the bicycle demand forecasting model, and the two-layer GRU model is used as the second model of the bicycle demand forecasting, and the two levels of models are combined to form our overall forecasting model. The overall model framework is shown in Fig. 5. In the figure, RF denotes Random Forest model, BG denotes Bagging model, LGB denotes LightGBM model and XGB denotes XGBoost model. The weights for the weighted average fusion are determined according to the entropy weighting method [26]. The mathematical expressions can be denoted as

$$x_{ij}^{'}=\frac{|x_{ij}-\min(x_j)|}{\max(x_j)-\min(x_j)} \tag{5}$$

$$e_j=-\frac{1}{\ln(n)}\sum_{i=1}^{n}\frac{x_{ij}^{'}}{\sum_{i=1}^{n}x_{ij}^{'}}\ln\left(\frac{x_{ij}^{'}}{\sum_{i=1}^{m}x_{ij}^{'}}\right),j=1,\ldots,m \tag{6}$$

$$\omega_j=\frac{1-e_j}{\sum_{j=1}^{m}(1-e_j)},j=1,\ldots,m \tag{7}$$

$$s_j=\frac{\sum_{j=1}^{m}\omega_j x_{ij}}{\sum_{j=1}^{m}\omega_j},i=1,\ldots,n \tag{8}$$

x_{ij} refers to the ith prediction result of the jth model, and $x_{ij}^{'}$ is the result obtained after normalizing x_{ij}. i ranges from 1 ~ n and j ranges from 1 ~ m. e_j represents the information entropy of the jth model, ω_j represents the weight of the jth model, and s_j represents the result obtained after weighted average fusion.

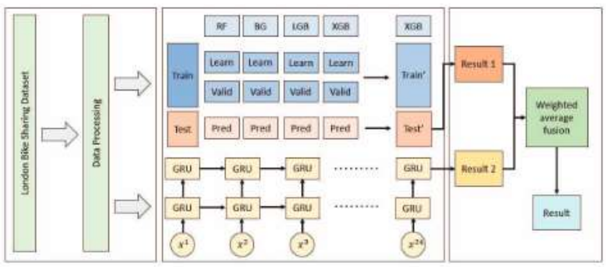

Fig. 5. GRU-Stacking based bicycle demand prediction model framework

F. Evaluation

This article selects Linear regression coefficient of determination (R^2), Root Mean Squared Logarithmic Error (RMSLE) and Mean Absolute Percentage Error (MAPE) to measure the performance of the different models. They are denoted by

$$R^2=\frac{\sum_{p=1}^{n}\left(\hat{y}_p-\bar{y}\right)^2}{\sum_{p=1}^{n}\left(y_p-\bar{y}\right)^2} \tag{9}$$

$$\text{RMSLE}=\sqrt{\frac{1}{n}\sum_{p=1}^{n}\left(\log\left(y_p+1\right)-\log\left(\hat{y}_p+1\right)\right)^2} \tag{10}$$

$$\text{MAPE}=\frac{1}{n}\sum_{p=1}^{n}\left|\frac{\hat{y}_p-y_p}{y_p}\right| \tag{11}$$

Where: n denotes sample size, y_p denotes sample real value, $\hat{y}_p$ denotes sample anticipated value, and $\bar{y}$ denotes sample mean.

IV. Results

A. Filtering the base model and meta-model in Stacking

We use the four machine learning models XGBoost, LightGBM, Random Forest, and Bagging as the base models for the Stacking integration method, and then take one of the four models as the metamodel. To determine the ideal hyperparameter values for each of the four machine learning models, we employed the grid parameter search technique offered by Sklearn. For instance, following parameter modification, the values of the parameter max length, which is shared by XGBoost, LightGBM, and Random Forest, are 10, 12, and None for the three models, respectively.

We can see from the findings in TABLE I that the base model of XGBoost, LightGBM, Bagging, and Random Forest combined with XGBoost as the meta-model beats the other base and meta-model combinations. As a result, we decided to employ the model created utilizing the aforementioned integration techniques as the first level model for our complete prediction model.

TABLE I. Results of Stacking integration with different combinations of base models and metamodels

Base model	Meta model							
	XGB		*LGB*		*RF*		*BG*	
	RMLSE	R^2	*RMLE*	R^2	*RMLSE*	R^2	*RMLSE*	R^2
RF+XGB	0.274	0.918	0.274	0.925	0.288	0.920	0.292	0.914
BG+LGB	0.276	0.931	0.280	0.933	0.290	0.930	0.297	0.928
BG+XGB	0.270	0.920	0.275	0.924	0.285	0.922	0.291	0.920
LGB+XGB	0.270	0.931	0.272	0.935	0.283	0.929	0.290	0.928
RF+BG	0.285	0.920	0.286	0.925	0.297	0.919	0.301	0.916
RF+LGB	0.273	0.931	0.277	0.933	0.286	0.931	0.301	0.916
LGB+XGB+RF	0.268	0.931	0.271	0.934	0.276	0.932	0.281	0.929
LGB+XGB+BG	0.269	0.931	0.271	0.934	0.278	0.932	0.284	0.931
XGB+RF+BG	0.273	0.920	0.275	0.926	0.280	0.921	0.286	0.923
LGB+RF+BG	0.274	0.931	0.273	0.925	0.284	0.921	0.286	0.923
XGB+LGB+BG+RF	0.267	0.936	0.270	0.936	0.274	0.932	0.280	0.931

B. Performance of different models on the bike-sharing demand forecasting problem

We compare the evaluation results of the GRU-Stacking prediction model constructed with XGBoost, LightGBM, LCE [27], the first-level model constructed in this paper through the Stacking integration method, Bi-GRU and Bi-GRU for the bicycle sharing demand prediction problem. It is important to note that the two-layer LSTM and two-layer GRU models we selected are both two-layer unidirectional models since, according to our tests, they outperform all other two-layer models. In addition, the parameter values for the XGBoost model, the LightGBM model, and the first-level model built using the Stacking integration approach are the same as in Section IV.A, and the crucial parameter settings for the GRU and LSTM are displayed in TABLE II.

TABLE II. Important hyperparameter settings for GRU and LSTM

Hyperparameter	*LSTM*	*GRU*
sequence length	0.901	0.273
epochs	0.934	0.303
batch size	0.902	0.320

The final evaluation findings for the various models on the demand dataset for bicycle sharing are provided in TABLE III, Stacking represents XGBoost, LightGBM, Bagging and Random Forest as the base model and XGBoost as the meta model. In terms of R^2, RMSLE and MAPE scores, our GRU-Stacking model performs better than the other models, notably in the RMSLE and MAPE assessment metrics, which are noticeably higher than the other models, reaching 0.209 and 0.142 respectively. This shows that our model performs well when used to anticipate demand for bicycle sharing.

TABLE III. Comparison of evaluation results of different models on the bike-sharing demand dataset

Models	R^2	*RMSLE*	*MAPE*
XGB	0.901	0.273	0.202
LGB	0.934	0.303	0.195
LCE	0.902	0.320	0.219
Stacking	0.932	0.267	0.185
GRU	0.955	0.273	0.201
Bi-GRU	0.956	0.288	0.225
Two-layer GRU	0.966	0.253	0.157
LSTM	0.966	0.298	0.204
Bi-LSTM	0.965	0.295	0.207
Two-layer LSTM	0.964	0.254	0.178

Models	*R^2*	*RMSLE*	*MAPE*
XGB	0.901	0.273	0.202
GRU-Stacking	0.968	0.209	0.143

V. CONCLUSION

We propose a GRU-Stacking based bicycle sharing demand prediction model, based on three evaluation metrics, R^2, RMSLE and MAPE, which all outperform the more mainstream models such as XGBoost, LCE, Bi-LSTM and Bi-GRU on the London Bike Sharing Dataset. This demonstrates that this model can be well used for bicycle sharing prediction problems. In the future, we can use this model for other problems of data information prediction. In addition, we plan to extend our prediction model by incorporating more advanced models and investigate the impact of various fusion methods to achieve better prediction results.

REFERENCES

[1] Li W , Kamargianni M . Providing quantified evidence to policy makers for promoting bike-sharing in heavily air-polluted cities: A mode choice model and policy simulation for Taiyuan-China[J]. Transportation Research Part A Policy & Practice, 2018, 111(MAY):277-291.Transportation research part A: policy and practice, 2018, 111: 277-291.

[2] EEA. The European Environment: State and Outlook 2005. 2010.

[3] Zhang L , Zhang J , Duan Z Y , Bryde D. Sustainable bike-sharing systems: characteristics and commonalities across cases in urban China[J]. Journal of Cleaner Production, 2015, 97(jun.15):124-133.

[4] Bauman A , Crane M , Drayton B A , Titze S. The unrealised potential of bike share schemes to influence population physical activity levels - A narrative review[J]. Preventive Medicine, 2017:S0091743517300713.

[5] Bo W A , Ik A . Short-term prediction for bike-sharing service using machine learning[J]. Transportation Research Procedia, 2018, 34:171-178.

[6] Choi S H, Han M K. The empirical evaluation of models predicting bike sharing demand[C]//2020 International Conference on Information and Communication Technology Convergence (ICTC). IEEE, 2020: 1560-1562.

[7] Zhou Z H. Ensemble learning[M]//Machine learning. Springer, Singapore, 2021: 181-210.

[8] Shi X , Chen Z , Wang H , Yeung DY , Wong WK , Woo WC. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting[C]// MIT Press. MIT Press, 2015.

[9] Cho K, Van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.

[10] Eren E , Uz V E . A Review on Bike-Sharing: The Factors Affecting Bike-Sharing Demand[J]. Sustainable Cities and Society, 2019, 54:101882.

[11] Campbell A A, Cherry C R, Ryerson M S, Yang X. Factors influencing the choice of shared bicycles and shared electric bikes in Beijing[J]. Transportation research part C: emerging technologies, 2016, 67: 399-414.

[12] Mattson J , Godavarthy R . Bike Share in Fargo, North Dakota: Keys to Success and Factors Affecting Ridership[J]. Sustainable Cities and Society, 2017, 34:174–182.

[13] Boufidis N , Nikiforiadis A , Chrysostomou K , et al. Development of a station-level demand prediction and visualization tool to support bike-sharing systems' operators[J]. Transportation Research Procedia, 2020, 47:51-58.

[14] Mehta S , Patnaik K S . Stacking Based Ensemble Learning for Improved Software Defect Prediction[J]. 2021.

[15] Fu H , Gao S , Peng Y , et al. Prediction of fishing vessel operation mode based on Stacking model fusion[J]. Journal of Physics Conference Series, 2021, 1792(1):012030.

[16] Wang S, Cao J, Yu P. Deep learning for spatio-temporal data mining: A survey[J]. IEEE transactions on knowledge and data engineering, 2020.

[17] Miao Y , Han J , Gao Y , Zhang B. ST-CNN: Spatial-Temporal Convolutional Neural Network for Crowd Counting in Videos[J]. Pattern Recognition Letters, 2019, 125(JUL.):113-118.

[18] Yan S , Xiong Y , Lin D . Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition[C]// 2018.

[19] Shahid F , Zameer A , Muneeb M . Predictions for COVID-19 with Deep Learning Models of LSTM, GRU and Bi-LSTM[J]. Chaos Solitons & Fractals, 2020:110212.

[20] Wang S , Zhao J , Shao C, Dong C, Yin C. Truck Traffic Flow Prediction Based on LSTM and GRU Methods With Sampled GPS Data[J]. IEEE Access, 2020, 8:208158-208169.

[21] BreimanLeo. Bagging predictors[J]. Machine Learning, 1996.

[22] Breiman L. Random forests[J]. Machine learning, 2001, 45(1): 5-32.

[23] Chen T, Guestrin C. Xgboost: A scalable tree boosting system[C]//Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016: 785-794.

[24] Qi M . LightGBM: A Highly Efficient Gradient Boosting Decision Tree[C]// Neural Information Processing Systems. Curran Associates Inc. 2017.

[25] Chung J , Gulcehre C , Cho K H , Bengio Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling[J]. Eprint Arxiv, 2014.

[26] Zhao J, Ji G, Tian Y, Chen Y, Wang Z. Environmental vulnerability assessment for mainland China based on entropy method[J]. Ecological Indicators, 2018, 91: 410-422.

[27] Fauvel K, Masson V, Fromont E, Faverdin P, Termier A. Towards sustainable dairy management-a machine learning enhanced method for estrus detection[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019: 3051-3059.