Performance Comparison of Axle Weight Prediction Algorithms on Time-Series Data

Žiga Kolar
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
ziga.kolar@ijs.si

David Susič
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
david.susic@ijs.si

Martin Konečnik
Cestel Cestni Inženiring d.o.o
Špruha 32
Trzin, Slovenia
martin.konecnik@cestel.si

Domen Prestor
Cestel Cestni Inženiring d.o.o
Špruha 32
Trzin, Slovenia
domen.prestor@cestel.si

Tomo Pejanovič Nosaka
Cestel Cestni Inženiring d.o.o
Špruha 32
Trzin, Slovenia
tomo.pejanovic@cestel.si

Bajko Kulauzović
Cestel Cestni Inženiring d.o.o
Špruha 32
Trzin, Slovenia
bajko@cestel.si

Jan Kalin
Zavod za gradbeništvo Slovenije
Dimičeva ulica 12
Ljubljana, Slovenia
jan.kalin@zag.si

Matjaž Skobir
Cestel Cestni Inženiring d.o.o
Špruha 32
Trzin, Slovenia
matjaz.skobir@cestel.si

Matjaž Gams
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
matjaz.gams@ijs.si

Abstract

Accurate vehicle axle weight estimation is essential for the maintenance and safety of transportation infrastructure. This study evaluates and compares the performance of various algorithms for axle weight prediction using time-series data. The algorithms assessed include traditional machine learning models (e.g., random forest) and advanced deep learning techniques (e.g., convolutional neural networks). The evaluation utilized datasets comprising time-series data from 10 sensors positioned on a single lane of a bridge, with the goal of predicting each vehicle's axle weights based on the signals from these sensors. Each algorithm's performance was measured against the OIML R-134 recommendation, where a prediction was classified as accurate if the error was within ±4 percent for two-axle vehicles and ±8 percent for vehicles with more than two axles. Tests were conducted on several bridges, with this paper presenting detailed results from the Lopata bridge. Findings indicate that deep learning models, particularly convolutional neural networks, significantly outperform traditional methods in terms of accuracy and their ability to adapt to complex patterns in time-series data. This study provides a valuable reference for researchers and practitioners aiming to enhance axle weight prediction systems, thereby contributing to more effective infrastructure management and safety monitoring.

Keywords

time-series data, axle weight, machine learning, neural network

1 Introduction

Accurate axle weight prediction plays a pivotal role in the maintenance and safety of transportation infrastructure [7]. The precise estimation of axle weights is essential for various applications, including road maintenance planning, traffic management, and the prevention of overloading, which can lead to premature road wear and increased accident risks [8]. Traditional methods for axle weight measurement often rely on static scales or weigh-in-motion (WIM) systems. While these methods provide direct measurements, they are susceptible to limitations such as high installation and maintenance costs, potential measurement inaccuracies due to environmental factors, and the need for frequent calibration.

In recent years, the advent of advanced computational techniques has opened new avenues for improving axle weight prediction. Machine learning (ML) and deep learning (DL) algorithms, in particular, offer promising alternatives by leveraging time-series data to model complex, non-linear relationships inherent in vehicular weight patterns. These methods can enhance prediction accuracy, handle large volumes of data, and adapt to varying conditions, making them suitable for real-world applications where traditional methods may fall short.

This study systematically evaluates and compares the performance of various axle weight prediction algorithms using time-series data. We focus on a diverse set of algorithms, including machine learning models like random forests (RF) [6] and advanced deep learning techniques such as convolutional neural networks (CNN) [4].

The objective of this research is to explore the potential of combining traditional WIM systems with advanced ML and DL models to enhance axle weight predictions. By comparing the performance of different methodologies, including the SIWIM traditional model, random forest (IJS RF), a hybrid approach (AVERAGE(IJS, SIWIM traditional)), and a CNN-based model, this study aims to identify the most effective strategies for accurate and reliable axle weight estimation. Additionally, it examines the impact of synthetic data generation on the performance of these models, providing a comprehensive evaluation of their practical applicability in real-world scenarios.

The study aimed to predict the axle weights of vehicles using ten input signals from sensors placed under the Lopata bridge.

Information Society 2024, 7–11 October 2024, Ljubljana, Slovenia

https://doi.org/10.70314/is.2024.scai.4752

Each predictive algorithm's performance was evaluated according to the OIML R-134 recommendation, which is deemed accurate if the error margin for predicting the axle weight is within ±4% for vehicles with two axles and within ±8% for vehicles with more than two axles.

The dataset comprised 1478 samples, i.e. passing of a vehicle, each containing 10 signals per vehicle. For each sample, a static weight for each axle was assigned as the target value. Static weight refers to the weight measured by a scale when the vehicle is stationary.

This paper is structured as follows: Section 2 reviews several state-of-the-art approaches. Section 3 details the preprocessing steps necessary before applying machine learning methods. In Section 4, algorithms used for predicting axle weights are presented. Section 5 presents the final results of the axle weight predictions. Finally, Section 6 summarizes the findings and proposes ideas for future research.

2 Related Work

The prediction of axle weights using time-series data has often been studied in recent years, resulting in a substantial body of related work. Below, several state-of-the-art (SOTA) approaches are described.

Zhou et al. [10] differentiated between high-speed and low-speed weigh-in-motion (WIM) systems and analyzed the characteristics of axle weight signals. They proposed a nonlinear curve-fitting algorithm, detailing its implementation. Numerical simulations and field experiments assessed the method's performance, demonstrating its effectiveness with maximum weighing errors for the front axle, rear axle, and gross weights recorded at 4.01%, 5.24%, and 3.92%, respectively, at speeds of 15 km/h or lower.

Wu et al. [8] introduced a modified encoder-decoder architecture with a signal-reconstruction layer to identify vehicle properties (velocity, wheelbase, axle weight) using the bridge's dynamic response. This unsupervised encoder-decoder method extracts higher features from the original data. A numerical bridge model based on vehicle-bridge coupling vibration theory demonstrated the method's applicability. Results indicated that the proposed approach accurately predicts traffic loads without additional sensors or vehicle weight labels, achieving better stability and reliability even with significant data pollution.

Xu et al. [9] applied wavelet transform for denoising and reconstructing the WIM signal, and used a back propagation (BP) neural network optimized by the brain storm optimization (BSO) algorithm to process the WIM signal. Comparing the predictive abilities of BP neural networks optimized by different algorithms, they found the BSO-BP WIM model to exhibit fast convergence and high accuracy, with a maximum gross weight relative error of 1.41% and a maximum axle weight relative error of 6.69%.

Kim et al. [5] developed signal analysis algorithms using artificial neural networks (ANN) for Bridge Weigh-in-Motion (B-WIM) systems. Their procedure involved extracting information on vehicle weight, speed, and axle count from time-domain strain data. ANNs were selected for their effectiveness in incorporating dynamic effects and bridge-vehicle interactions. Vehicle experiments with various load cases were conducted on two bridge types: a simply supported pre-stressed concrete girder bridge and a cable-stayed bridge. High-speed and low-speed WIM systems were used to cross-check and validate the algorithms' performance.

Bosso et al. [1] proposed a method using weigh-in-motion (WIM) data and regression trees to identify patterns in overloaded truck weights and travel. The analysis reveals that truck type is the key predictor of overloading, while time of day is crucial for axle overloading, with most incidents occurring late at night or early morning. These insights can enhance enforcement strategies and inform pavement management and design, optimizing infrastructure longevity and safety.

He et al. [2] introduced a new method that uses only the flexural strain signals from weighing sensors to identify axle spacing and weights, reducing installation costs and expanding BWIM applications. The method's accuracy is validated through numerical simulations and laboratory experiments with a scaled vehicle-bridge interaction model, showing promising results for accurate axle spacing and weight identification.

3 Data Preprocessing

Before applying various algorithms to the dataset, several preprocessing steps were necessary. Due to the differing lengths of signals from each sample, padding was performed to standardize them to the length of the longest signal. Samples with a gross weight below 5 kN were excluded from both the training and test datasets. Each signal was cropped by removing data to the left of the leftmost peak value minus 100 and to the right of the rightmost peak value plus 100. The peak values were calculated in advance.

To address the limited availability of data required for deep learning, which typically necessitates tens of thousands of samples for effective training, synthetic data generation was employed. The original dataset comprised 1,478 samples (from January 2022 to December 2023) i.e. passing of a vehicle, each containing 10 signals per vehicle. An additional 20,000 synthetic samples were generated using a specific algorithm. This algorithm operates by iterating 20,000 times, during each of which a random training sample and a random strain factor were selected. The strain factor is a random value ranging between 0.5 and 0.99. The selected signal from the training sample was then scaled by the chosen strain factor. This scaling process effectively models the feature that doubling the amplitude of the signal corresponds to doubling its weight.

A crucial aspect of data preprocessing involved the normalization of sensor signals to ensure uniformity across the dataset. Each signal was normalized to have a mean of zero and a standard deviation of one, which helps in improving the convergence of machine learning algorithms by ensuring that each feature contributes equally to the learning process.

The selection of training and test data was conducted using a rolling window approach [3]. Specifically, for each testing month, the training data comprised all available data up to but not including the testing month. For instance, if May 2023 was designated as the testing month, the training dataset consisted of data from January 2022 through April 2023. This process was systematically repeated for each testing month from March 2022 to December 2023.

4 Methodology

Four methods were identified as applicable for predicting vehicle axle weights. The first method, known as SIWIM traditional [11], calculated the number of axles, axle lengths, and axle weights by utilizing influence lines to model the signal and determine the correct output. For validation purposes, each predicted output

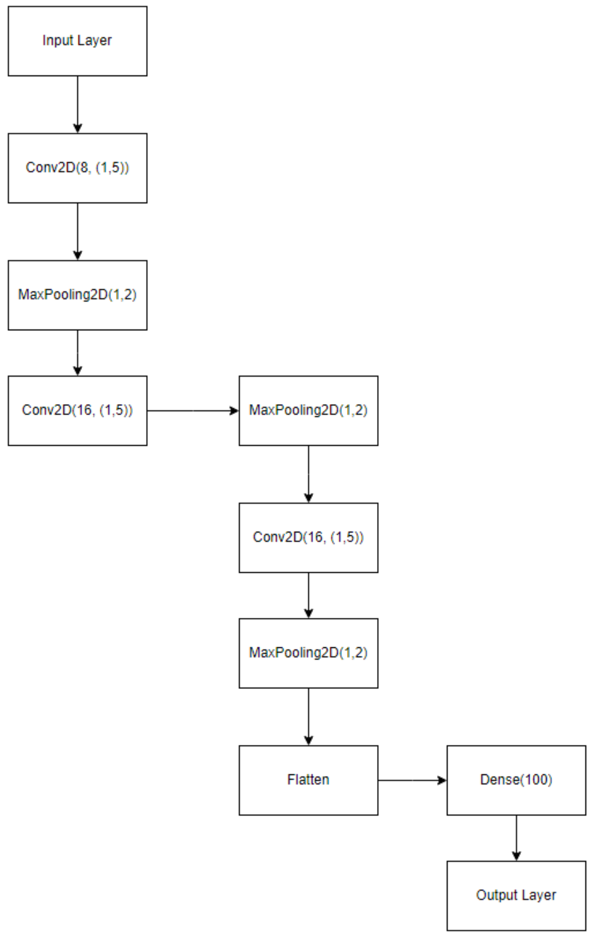

Figure 1: Architecture of CNN for predicting axle weights.

was stored in a separate file alongside the signal data, enabling direct comparison with the actual values.

The second method used the random forest [6] (named IJS RF) for predicting vehicle axle weights. The model relied on accurately identifying the positions of peaks to function correctly. Peak values were determined using the $find\_peaks$ method from the SciPy library, which identifies peaks based on a specified minimum height. Once the peaks were identified, the algorithm extracted values within a ±5 range of each peak. These extracted values were then used as input features for the random forest model. Additionally, the random forest model incorporated temperature, axle distances and gross weight as input features. Random forest algorithms are not inherently suited for time series data; however, they perform effectively with numerical data such as temperature, axle distance, and gross weight. Therefore, this algorithm was chosen for analyzing this type of input data.

The third method integrated the first two approaches by averaging the outputs from the SIWIM traditional and IJS RF models (named AVERAGE(IJS, SIWIM traditional)). This approach is motivated by the concept that combining multiple models can often yield more accurate results than relying on a single model alone [12].

The final method employed a convolutional neural network (CNN) to predict axle weights. The CNN utilized synthetic data, as detailed in section 3, during the training phase. This method processed all 10 signals as input to calculate the axle weights. The detailed architecture of the CNN is shown in Figure 1. 2D Convolutional layers (Conv2D) were used instead of 1D Convolutional layers due to the input data consisting of 10 sensor signals. The number of filters and kernel size are specified within the parentheses of each Conv2D layer, while the pooling size is defined in each 2D MaxPooling layer parentheses (MaxPooling2D). The last Dense layer has 100 neurons. To mitigate overfitting, a Dropout layer was added after the final Dense layer. Additionally, Batch Normalization was applied after each 2D Convolutional layer to further reduce the risk of overfitting.

Although Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) neural networks could be used for this task, a Convolutional Neural Network (CNN) was chosen instead because of its strengths in capturing spatial hierarchies and local patterns within the data. CNNs are highly effective at extracting local features and detecting patterns, while LSTM and GRU are better suited for handling temporal dependencies, which are not that relevant to this specific task.

5 Results

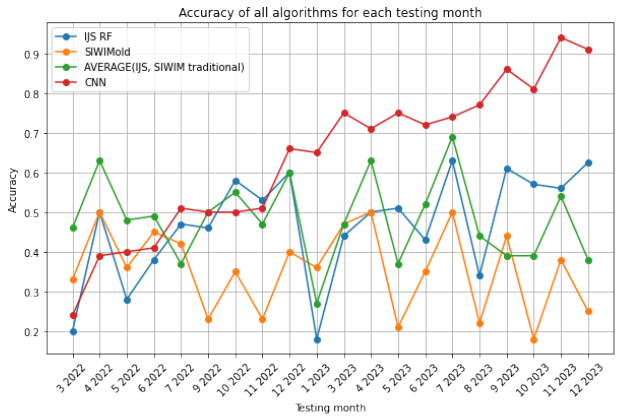

Figure 2: Accuracies of all algorithms for each testing month.

The results of each method described in Section 4 are illustrated in Figure 2. Among the methods evaluated, SIWIM traditional exhibited the poorest performance, with fluctuating trends observed throughout the entire two-year period. The CNN began to outperform the other three approaches after December 2022. Conversely, the AVERAGE(IJS, SIWIM traditional) method showed superior performance during the initial testing months from March 2022 to June 2022.

The performance of the CNN improved with an increasing amount of data, whereas the IJS RF and AVERAGE(IJS, SIWIM traditional) methods were more effective during the initial phase when less training data was available. However, the improvement in CNN's accuracy was not linear. This non-linear trend can be attributed to the random initialization of the CNN's weights before each training session, occasionally leading to suboptimal convergence.

An additional analysis was conducted to compare the performance of the models under varying environmental conditions, such as temperature fluctuations and differing traffic patterns. This analysis revealed that the CNN model maintained its accuracy more consistently across different conditions, indicating its robustness and adaptability. Furthermore, the inclusion of synthetic data in training the CNN model contributed to its superior performance, as it allowed the model to learn from a more diverse set of examples. Future research should focus on expanding the range of synthetic data and exploring additional ensemble techniques to further enhance prediction accuracy.

Despite achieving high accuracy with the CNN model, with the highest accuracy reaching 0.94, this most accurate method still falls short of meeting the OIML R-134 recommendation by 4.4%. Furthermore, the results show that more static data could be needed for the learning phase. Having 1000 static samples which were augmented might not be sufficient to reach the OIML R-134 recommendation.

In summary, the results indicate that while traditional methods such as IJS RF and AVERAGE(IJS, SIWIM traditional) perform well with limited data, convolutional neural networks (CNNs) demonstrate superior performance as more data becomes available, despite some variability in their convergence. In addition, a sufficient number of training examples is needed to approach the desired OIML R-134 recommendation.

6 Conclusion and Discussion

In this study, a performance comparison of various axle weight prediction algorithms using time-series data collected from 10 sensors positioned on the Lopata bridge was conducted. The algorithms evaluated encompassed traditional machine learning models, such as random forests, and advanced deep learning techniques, notably convolutional neural networks.

The major findings reveal that CNNs achieved significantly better results in predicting axle weights during the latter months of the experiment. The CNNs' ability to adapt to and learn from complex patterns within the time series data was a key factor in their superior performance. Despite achieving high accuracy with the CNN model, reaching a peak accuracy of 0.94, this method still falls short of meeting the OIML R-134 recommendation by 4.4%.

Overall, there are three implications of this study. First, it demonstrates the potential of deep learning techniques to enhance the accuracy of axle weight predictions where sufficient data is available, thereby facilitating more reliable infrastructure management. Second, for smaller datasets, it is more effective to use classical machine learning systems in combination with methods like SIWIM traditional. Third, it provides a valuable benchmark for researchers and practitioners, guiding the development and implementation of more effective axle weight prediction systems.

To achieve the OIML R-134 recommendation, two options are possible:

- Just add more data - if the trend continues, adding another half a year of measurements would enable achieving the standard. Another option would be to apply measurements on a bridge with more traffic.
- Improve the methods by incorporating advanced ensemble techniques.

To introduce the ensemble approaches, one potential improvement involves modeling each sensor individually. This approach entails building a separate CNN model for each of the ten sensors, allowing for more specialized and potentially more accurate predictions from each sensor's data. By focusing on the unique characteristics and data patterns of each sensor, the models can be better tailored to capture specific nuances in the time-series data.

After developing individual models for each sensor, the next step would be to combine the predictions from these models into a single final prediction. This can be achieved using an ensemble method, such as a random forest classifier. The random forest classifier would take the ten individual predictions (one from each sensor model) as input features and produce a consolidated final axle weight prediction.

This method not only holds the potential to improve the accuracy and robustness of the axle weight predictions but also provides a scalable framework that can be adapted to different datasets and sensor configurations. Future work should explore the implementation of this approach, including the optimization of individual sensor models and the integration of their predictions through an ensemble method.

By advancing the CNN model in this manner, it is anticipated that the performance gap relative to the OIML R-134 recommendation could be further reduced, bringing the predictions closer to the required accuracy levels with a smaller amount of data and enhancing the overall efficacy of the axle weight prediction system.

Acknowledgements

This study received funding from company Cestel. The authors acknowledge the funding from the Slovenian Research and Innovation Agency (ARIS), Grant (PR-10495) and Basic core funding P2-0209. The author(s) made use of chatGPT to assist with this article. ChatGPT was commonly employed as a tool for enhancing the language of the initial draft, without altering the length of the text. ChatGPT 4 was accessed/obtained from chatgpt.com and used with modification in July 2024.

References

[1] Mariana Bosso, Kamilla L Vasconcelos, Linda Lee Ho, and Liedi LB Bernucci. 2020. Use of regression trees to predict overweight trucks from historical weigh-in-motion data. *Journal of Traffic and Transportation Engineering (English Edition)*, 7, 6, 843–859.

[2] Wei He, Tianyang Ling, Eugene J OBrien, and Lu Deng. 2019. Virtual axle method for bridge weigh-in-motion systems requiring no axle detector. *Journal of Bridge Engineering*, 24, 9, 04019086.

[3] Hamed Kalhori, Mehrisadat Makki Alamdari, Xinqun Zhu, Bijan Samali, and Samir Mustapha. 2017. Non-intrusive schemes for speed and axle identification in bridge-weigh-in-motion systems. *Measurement Science and Technology*, 28, 2, 025102.

[4] Teja Kattenborn, Jens Leitloff, Felix Schiefer, and Stefan Hinz. 2021. Review on convolutional neural networks (cnn) in vegetation remote sensing. *ISPRS journal of photogrammetry and remote sensing*, 173, 24–49.

[5] Sungkon Kim, Jungwhee Lee, Min-Seok Park, and Byung-Wan Jo. 2009. Vehicle signal analysis using artificial neural networks for a bridge weigh-in-motion system. *Sensors*, 9, 10, 7943–7956.

[6] Steven J Rigatti. 2017. Random forest. *Journal of Insurance Medicine*, 47, 1, 31–39.

[7] Mohhammad Sujon and Fei Dai. 2021. Application of weigh-in-motion technologies for pavement and bridge response monitoring: state-of-the-art review. *Automation in Construction*, 130, 103844.

[8] Yuhan Wu, Lu Deng, and Wei He. 2020. Bwimnet: a novel method for identifying moving vehicles utilizing a modified encoder-decoder architecture. *Sensors*, 20, 24, 7170.

[9] Suan Xu, Xing Chen, Yaqiong Fu, Hongwei Xu, and Kaixing Hong. 2022. Research on weigh-in-motion algorithm of vehicles based on bso-bp. *Sensors*, 22, 6, 2109.

[10] ZF Zhou, P Cai, and RX Chen. 2007. Estimating the axle weight of vehicle in motion based on nonlinear curve-fitting. *IET science, measurement & technology*, 1, 4, 185–190.

[11] A Žnidarič, J Kalin, M Kreslin, M Mavrič, et al. 2016. Recent advances in bridge wim technology. In *Proc. 7th International Conference on WIM.*

[12] Hui Zou and Yuhong Yang. 2004. Combining time series models for forecasting. *International journal of Forecasting*, 20, 1, 69–84.