Towards an AI-based early warning system for bridge scour

Negin Yousefpour a and Oscar Correa b

aDepartment of Infrastructure Engineering, The University of Melbourne, Parkville, VIC, Australia; bSchool of Computing and Information Systems, The University of Melbourne, Parkville, VIC, Australia

ABSTRACT

Scour is the number one cause of bridge failure in many parts of the world. Considering the lack of reliability in existing empirical equations for scour depth estimation and the complexity and uncertainty of scour as a physical phenomenon, it is essential to develop more reliable solutions for scour risk assessment. This study introduces a novel AI approach for early forecast of scour based on real-time monitoring data obtained from sonar and stage sensors installed at bridge piers. Long-Short Term Memory Networks (LSTMs), a prominent Deep Learning algorithm successfully used for time-series forecasting in other fields, were developed and trained using river stage and bed elevation readings for more than 11 years, obtained from Alaska scour monitoring programme. The capability of the AI models in scour prediction is shown for three case-study bridges. Results show that LSTMs can capture the temporal and seasonal patterns of both flow and river bed variations around bridge piers, through cycles of scour and filling and can provide reasonable predictions of upcoming scour depth as early as seven days in advance. It is expected that the proposed solution can be implemented by transportation authorities for development of emerging AI-based early warning systems, enabling superior bridge scour management.

1. Introduction

Scour at bridge piers has been a global challenge. It is the number one cause of bridge failure in many countries. With more than 26,000 scour critical bridges in the US, scour accounts for around 60% of bridge failures (Wardhana and Hadipriono 2003). In Australia, many bridge failures have been reported in recent years, mainly due to extreme weather conditions, in particular flooding (Lebbe et al. 2014). The annual probability of a scour-related bridge failure in the UK has been estimated at around 30% (Dikanski et al. 2016; Van Leeuwen and Lamb 2014). In Taiwan, in 2009 alone, approximately 150 bridges collapsed during extreme flooding (Hong et al. 2012).

Scour is known as a highly uncertain physical phenomenon due to complexity of flow, structure and soil interaction. Numerous empirical scour models have been developed in the past few decades, calibrated based on laboratory experimentation (Liang et al. 2017; Sheppard, Odeh, and Glasser 2004; Wang, Yu, and Liang 2017). Despite the research efforts, assessment of existing empirical scour models shows that the majority can result in significant errors, in particular over-prediction of scour. This is mainly because these models were calibrated with limited laboratory tests under certain conditions without field verification (Mueller and Wagner 2005; Sheppard, Melville, and Demir 2014). The most commonly used scour equation for bridge design is the HEC-18, which has evolved throughout the past two decades (Arneson et al. 2012). A recent review of the HEC-18 model by Schuring, Dresnack, and Golub (2017) reveals several limitations, the most important of which are (I) insensitivity to a broad range of geological conditions, especially very coarse granular particles, (II) impracticality of proposed solutions for cohesive soils, (III) lack of recommendations for future scour potential for existing bridges, i.e. field performance over the time.

To date, numerous research studies have worked on development of scour predictive models using “classical” Machine Learning (ML), as opposed to “modern” ML or Deep Learning (DL), trained based on both lab and field data (Azamathulla et al. 2010; Ebtehaj et al. 2016; Kim, Fard, and Chattopadhyay 2015; Zounemat-Kermani et al. 2009). Sharafati et al. (2021) have provided a comprehensive review on the most recent ML models for scour and their merits and limitations. These models generally provide more accurate maximum scour depth predictions compared with the...
empirical scour models. However, like any classical ML model, their extrapolation capacity is limited and the accuracy of predictions decay significantly outside the convex hull of the training dataset, i.e. poor generalisation to unseen data outside the training data (McCarty, Haeringer, and Polilfe 2020; Yousefzadeh and Cao 2022). Moreover, these models are often high-dimensional, incorporating a large number of bridge, flow and soil characteristics as the predictive variables (input parameters) and are often trained using data compiled from various scattered locations, therefore only validated under certain data domain ranges. These factors further expose these models to extrapolation risk, rendering less reliability in scour predictions for an unseen bridge (Balestriero, Pesenti, and LeCun 2021).

In addition to limitations of the current scour prediction models, there are other factors to blame for scour failure, among which are: lack of proper design and analysis method for older bridges subjected to scour; hydraulic uncertainties, especially increase in flood intensity and climate change impacts (Khandel and Soliman 2019); and complexity of riverbed sediment and morphological behaviour (Lamb et al. 2017). Given the aforementioned limitations, bridge authorities have resorted to monitoring solutions to manage the scour risk. This includes both regular surveys and inspections as well as real-time monitoring using advanced sensor technologies.

Real-time scour monitoring have become increasingly prevalent in recent years and have enabled more reliable scour risk management for critical and large-scale bridges (Briaud et al. 2011; Lagasse et al. 1997; Prendergast and Gavin 2014; Transportation Research Board TRB). Chang et al. (2014) proposed a real-time monitoring solution using micro-cameras installed at bridge piers and advanced image processing and pattern recognition methods to monitor bed level variations in real-time. They validated the method through laboratory experiments and reported that image data reliably captured monitored scour-depth evolution. Maroni et al. (2020) proposed a scour hazard model for road and railway bridges, using Bayesian networks calibrated based on limited data from scour monitoring systems on a bridge network in Scotland. They reported the Bayesian network approach to significantly reduce the uncertainty in the scour depth assessment at unmonitored bridges. Lin et al. (2021) introduced a new scour monitoring technology using vibration-based arrayed sensors, Internet of Things (IoT) and artificial intelligence (AI). The water level changes around the bridge pier were captured using real-time CCTV images and processed using deep learning methods. The monitoring system was validated through laboratory flume experimentation and field measurements.

Yousefpour et al. (2020) investigated scour prediction using AI/ML, based on monitoring data in collaboration with USGS and some of the US departments of transportation (DOTs). Three ML methods for real-time forecast and prediction of maximum depth of scour were developed. The developed models showed promising results in scour depth prediction based on real-time monitoring data. The data included river bed elevation measurements through sonar sensors \( (y_{sonar}) \) and river stage (flood/water level) measurements through stage sensors \( (y_{stage}) \) installed on more than 20 bridges in Alaska between 2001 to 2017. The preliminary results showed great potential for Deep Learning (DL) using Long-short term memory networks (LSTMs) for real-time scour forecast.

Following this study, in this paper we introduce an innovative AI-based, early warning solution for scour assessment that can be integrated with the real-time scour monitoring systems, such as those installed for many bridges in various states in the USA (e.g. Alaska, Oregon, Colorado, Idaho). Novel LSTM algorithms are developed and trained to recognise the inherent scour and flood patterns in the historic monitoring data, i.e. time series of river stage and bed elevation. The AI models can process the monitoring data in real-time and generate intelligent forecasts on upcoming scour depth enabling a superior scour risk management system.

The essence of our research is creating a bridge-specific expert system for scour assessment, based on its historical and real-time sonar and stage sensor readings without the need to involve all the physical parameters typically used in scour equations, such as pier geometry, velocity and soil condition (particle size). We base our data-driven solution on the fact that DL models can learn complex physical processes from the data, without exposure to explicit physics knowledge of the process (Kiarashinejad et al. 2019; Schwartz 2021), while proving our hypothesis which is: the historic time-series of scour (bed elevation) and flow depth (water stage) and their cross-correlations embody the physical interactions of the soil, flow and structure for a given bridge, which can be assimilated by modern deep-learning methods to provide useful insights on upcoming scour events.

The theoretical bases for such hypothesis and the underlying assumptions lies with the scour physical process: (I) the DL models are developed to be bridge-specific, i.e. learning cannot be transferred to another bridge, therefore the physical parameters that are site-specific and not temporally variable, such as bridge/pier geometry and soil condition/geology are not incorporated; (II) The river stage variation with time can
implicitly incorporate the flow rate into the AI models. The flow discharge time-series show a high correlation with stage time-series, therefore flow velocity (discharge) is not considered as a direct input to the AI models (see the analyses under Impact of Flow Velocity); (III) the soil condition impacts on the scour process is implicitly encoded in the historic river bed and flow variation patterns.

Note that the aim of this research is not generating a ML model for maximum scour depth prediction, rather introducing an early-warning solution by integrating AI with the scour monitoring technology to provide predictive insights for risk assessments of upcoming scour events. The proposed solution is demonstrated in the following sections for three case study bridges in Alaska: Chilkat River (bridge ID: 742), Knik River (bridge ID: 539) and Sheridan Glacier No.3 (bridge ID: 230). These were the bridges with the most significant scour and filling associated with seasonal high flows and flood events and for which the sensor readings were consistently available.

### 2. Approach

#### 2.1. Scour monitoring data

The sonar and stage sensor readings have been provided by USGS and DOTs accompanied by short cover reports highlighting the major scour and flooding events, newly installed sensors and replacements or damages to sensors on monitored bridges, among other details. The real-time monitoring data for Alaska bridges can be accessed through the online monitoring platform available at USGS Alaska Science Center.

Prior to using the sensor data for model development, extensive pre-processing was performed to generate smooth continuous time series with synchronised steps between sonar and stage readings. In some cases, a small portion of data was missing due to issues with the sensors (damages due to debris, vandalism, etc.) or the readings were noisy and erroneous. Also, sensor reading intervals were not uniform across all the bridges and years of monitoring. The following main steps were taken for pre-processing of data. See (Yousefpour et al. 2020) for a full description of the data and pre-processing methods:

- Apparent bias shifts in the sensors were corrected based on manual inspection.
- Outliers were removed via median filtering.
- Timestamps were made uniform (1hr) throughout the dataset and synchronised between sonar and stage.
- Denoising/filtering technics including Moving Average and Low Pass Filter were applied to remove high-frequency noises and to capture the actual trend.
- Missing data were imputed using polynomial interpolation and Gaussian Process.

Figure 1 shows the map of the bridges with installed sensors across Alaska as part of the USGS real-time monitoring programme. Sounding survey results from
2001 to present with an image of the bridge with installed sensors for bridges 742, 539 and 230, are shown in Figures 2, 5 and 7 respectively. Also, Figures 3, 6 and 8 provide the raw versus processed data for recent years. Note that all the original data were provided in US units (ft for sensor readings of elevation). Exploring the data shows similar yearly periodicity across bridges in Alaska. Most bridges experience two major high flow/flooding around July/August and September/October. There is no reading due to frozen waters between November and March, hence the gaps (or straight lines) between the years.

2.1.1. Bridge 742 – Chilkat River

As observed from Figures 2 and 3, historical sounding and sensor data for bridge 742 shows cyclic scour and filling of up to 4 m due to seasonal high flows (floods) happening around July/August and September/October each year and the bed elevation fluctuating between 33 m to 37 m (see also Figure 19(a)). The sounding surveys allowed for cross-validating the readings from sonar and ensuring the maximum depth of scour is captured across the channel. Sonar has been installed on the third pier from the left bank, which is closest, among the nine piers, to the major scour hole across the river bed.
Figure 4 provides the histogram of the processed stage and sonar readings for bridge 742 from 2007 to 2017. The negative correlation between the two features can be observed in the joint histogram.

2.1.2. Bridge 539 – knik river
As shown in Figures 5 and 6, bridge 539 goes through cyclic scour and filling between summer and fall up to a maximum of 6m, with the river bed fluctuating between 4 m to 10 m elevations (see also Figure 20(a)). The sonar has been installed on the pier closest to the right bank.

Due to the 2006 major flood and damage to sensors, the Stage readings were unavailable in 2006.

2.1.3. Bridge 230 – sheridan glacier
Bridge 230 has gone through several over-topping episodes during major floods between 2001 to 2017, with a flood exceeding 100-year return period in 2006. As shown in Figures 7 and 8, the sonar has been installed on the pier closest to the right. The sonar recordings show up to 5 m cycles of scour and filling and bed elevation variation between 6 m to 11 m (see also Figure 21(a)).

Figure 3. Bridge 742 – sample of sensor data from continuous real-time monitoring: (a) top: processed; bottom: raw, (b) 2017 data – processed (line) and raw (dots).
Due to the sensors malfunction, the readings were not used in 2009 and 2010.

2.2. Early warning solution framework

Figure 9 presents a schematic of the AI early warning solution framework we propose in this paper. The first and second steps i.e. real-time monitoring of scour and pre-processing of the historic scour monitoring data were explained in the previous section. The third step is developing AI algorithms that learn from past to predict the future, i.e. the variation of bed elevation across a pre-defined forecast window. In the forth step, the upcoming maximum scour depth is estimated based on the probability distribution function (PDF) of predictions generated by the AI model. Finally, in the last step the predicted scour depth is evaluated with respect to a set of threshold defined based on failure risk assessments, considering the as-built design of the foundation. This process involves subtracting the predicted scour depth for a target probability of exceedance (e.g. < 10%) from the as-built embedment depth of the foundation to assess the residual bearing capacity. In this paper we focus on step-3, where we introduce LSTM networks as surrogate models for scour forecast as explained in the following sections.

2.3. Deep learning

One of the most prominent deep learning models to approach a time series forecasting problem is Recurrent Neural Networks (RNNs), capable of “remembering”. In addition to feed-forward connections between computational units (neurons), RNNs have recurrent connections where the output of a unit is fed back to itself with a weight and a time delay, which provides the algorithm with a memory of past activations. This is the backbone idea of the solution proposed in this paper.

In processing the time-series data, RNNs iterate through the sequential elements (time-steps) while maintaining a memory state of the activation at each step. The sequence, in this study, corresponds to a vector of two features: river stage (stage) and river bed elevation (sonar).

2.3.1. Long short-term memory networks

LSTM networks are special types of RNNs with an internal memory unit that can maintain short and long temporal patterns of past time-series data and incorporate them into future predictions (Deng and Yu 2014; Hochreiter and Schmidhuber 1997). Stacking memory units in such networks enables learning higher levels of temporal patterns in sequential data. LSTMs have been successfully applied in a wide range of applications, such as stock market forecast, text, language and voice recognition (Ray, Rajeswar, and Chaudhury 2015) and recently used in some geotechnical engineering applications (Xie, Zhou, and Chai 2019; Yousefpour and Pouragha 2022; Zhang et al. 2021, 2020).

One of the main advantages of LSTMs over simple RNNs is that the LSTM memory unit can overcome the vanishing gradient problem, as it allows the past information to be re-injected at a later time. Vanishing gradient is a common learning pitfall in deep recurrent neural networks, where the gradient of weights becomes too small or too large, if the network is unfolded for too many time steps (Bengio, Simard, and Frasconi 1994; Hochreiter and Schmidhuber 1997).

The LSTM algorithms in this study were developed using the TensorFlow library in Python. For time-intensive computations and configuration optimisation,
codes were run on several GPUs of Spartan, the high-performance computing system at The University of Melbourne. Figure 10 shows a diagram of the LSTM architecture used in this study.

To train the LSTMs, the time-series data are first sliced into smaller sequences through a sliding window. In each slice, a certain number of time-steps are treated as input (input width) to the network while a number of time-steps into the future is predicted (label width). The total length for each data sequence is the sum of the input and label widths (see Figure 11(a)). Note that no shift was considered in between the input and label chains in this study (offset width = label width).

For the training process, the sequences in each subset of data (training, validation and test) are divided into a number of batches. A maximum of 32 sequences are
selected for each batch (batch size = 32). The LSTM performance is evaluated via the mean squared error (loss) and the mean absolute error (MAE) of predictions over the label width for all the batches in a dataset. The network is trained using the training dataset over a large number of epochs (100) until the error (loss) on the validation dataset meets the stopping criteria. The most commonly activated stopping criteria is the threshold number for error increase recurrence (patience parameter). In addition to monitoring the training over the validation dataset, the Drop-out technique is also explored to overcome the over-fitting issue. The performance of the models for unseen data is evaluated through the test dataset.

Three LSTM variants were investigated as explained in the following sections:

1. Single-Shot (ss)
2. Feedback or Auto-regressive (fd)
3. Two-Layer, Single-Shot (ss2)

2.3.2. Single-Shot model
In this variant, all the time steps are predicted at once with no offset as depicted in Figure 11(b). The Python implementation of the model using TensorFlow comprises of three layers: an LSTM layer of units with sigmoid activation, a Dense layer, containing units with ReLU activation.

![Figure 6. Bridge 539 – Sample of sensor data from continuous real-time monitoring, (a) top: processed; bottom: raw, (b) 2016 data – processed (line) and raw (dots).](image-url)
activation function: $\max(0, x)$ and a Reshape layer, containing linear units to adjust the output dimensions at the end. The Dense layer has as many units as the number of time-steps to be predicted ($\text{label\_width}$) times the number of features ($\text{n\_Label\_Feat}$). The output from the Dense layer, which has two dimensions ($\text{batch}$...
size, label_width * n_LabelFeat), is reshaped into a tensor of three dimensions (batch size, label_width, n_LabelFeat), where the prediction of the label features is set up.

2.3.3. Feedback (Auto-regressive) model

Similar to Single-Shot, this variant also has one LSTM layer, however, instead of predicting the output time-steps all at once, predictions are made one step at a time. The hypothesis is whether feeding back the model’s output into itself at each step, i.e. conditioning prediction at each time-step on the previous one, increases the accuracy. As shown in Figure 11(c), the model predicts only one hour in the future and then that prediction is the input for the prediction of the next time-step.

An obvious advantage of this model is that the number of time-steps to be predicted in the future is unbounded.

2.3.4. Two-Layer, single-shot model

This model is a Single-Shot variant with two LSTM layers. It is sometimes useful to stack several LSTM layers to increase the representational power of a

---

**Figure 8.** Bridge 230 – sample of sensor data from continuous real-time monitoring, (a) top: processed; bottom: raw, (b) 2017 data – processed (line) and raw (dots).
network, as it is also common in feed-forward neural networks. The trade-off is between the computational cost and representational power of the model. Just to give an idea of how effective this strategy could be: Google Translate algorithms is a stack of seven large LSTM layers. It is generally a good idea to increase the network’s capacity, i.e. stacking layers, until over-fitting becomes an obstacle.

2.4. LSTM configurations and hyper-parameters

A number of LSTM configurations were analysed in this study, considering possible combination scenarios for input features, model variants and other hyper-parameters. Using the grid-search method, the most optimum models were identified by comparing numerous combinations of hyper-parameters. Table 1 presents the main hyper-parameters and their investigated values.

2.4.1. Features

The scour data has clear yearly periodicity (e.g. frozen waters, seasonal variations of river stage/flooding and scour/filling), therefore we incorporate two additional features representing “time of the year” by applying \( \sin(\cdot) \) and \( \cos(\cdot) \) functions to time-stamps. Hence, the two feature combinations are:

1. With the time features (ssy): \([\text{sonar, stage, year sin, year cos}]\)
2. Without time features (ss): \([\text{sonar, stage}]\)

2.4.2. Sliding window

The sliding-window hyper-parameter refers to the number of time-steps taken as input and label, i.e. input and label width. The following input-label widths were considered:

1. (336, 168): Two weeks of history, one week of predictions
2. (720, 168): Four weeks of history, one week of predictions
3. (720, 336): Four weeks of history, two weeks of prediction

Although initially increasing the input width may benefit the prediction accuracy, having long sequences will limit the number of sequences from the already small training dataset. A good trade-off between input
width and the number of sequences is one of the key elements of optimal training. With respect to the number of steps to be predicted (label width), the problem itself sets a minimum range, which is 168 to 336 hrs (one to two weeks). This is considered as the minimum required time for bridge authorities to take countermeasures and devise a plan of action to manage the scour risk (confirmed in a meeting with various DOT bridge managers).

### 2.4.3. Number of epochs

The number of epochs is set to 100, and the training was set to stop when monitored metrics stopped improving (the loss function stopped decreasing). The parameter
patience was set to 5, which is the number of consecutive epochs with no improvement, after which training is stopped.

2.4.4. Number of units

Similar to feed-forward neural networks, increasing the number of units (hidden neurons) increases the predictive capacity of the network, yet at the risk of over-fitting and increased computational cost. The chosen values for this parameter are powers of two: 32, 64, 128 and 256.

2.4.5. Regularisation: drop-out

The drop-out rate is the proportion of units whose inputs are zeroed out in order to break random correlations in the training data. The drop-out version for RNNs is called recurrent drop-out. Both drop-out and recurrent drop-out rates need to be optimally selected (Semeniuta, Severyn, and Barth 2016). Two values of 0 and 20% were considered in the grid-search.

**2.5. LSTMs as surrogate models for scour forecast**

Scour prediction as a multivariate time-series forecasting problem can be defined using the following mathematical representation. This also explains how the LSTM can act as a surrogate model for estimating the maximum scour depth:

Let’s consider an individual pier as shown in Figure 12, for an arbitrary bridge, where $y_i$ is the maximum scour depth, $y_1$ is the corresponding upstream flow depth and $y_{ab}$ is the as-built bed elevation at the bridge pier. Considering $y_{st}^i = \max {y_{st}^i}_{k=1}^n$ and $y_{so}^i = \min {y_{so}^i}_{k=1}^n$, the maximum and minimum of readings corresponding to the maximum flow and scour depth during the forecast window of $t = \{t_i\}_{i=1}^n = \{t_1, \ldots, t_{n-1}\}$, the AI forecast model using LSTM can be written as:

$$\chi\left(\left[\begin{array}{c} y_{st}^i \\ y_{so}^i \end{array}\right]\right)_{k=1}^{n-1} = \psi\left(\left[\begin{array}{c} y_{st}^i \\ y_{so}^i \end{array}\right]\right)_{m=1}^{n-1}, W, b$$  \hspace{1cm} (1)

where, $\left\{\left[\begin{array}{c} y_{st}^i \\ y_{so}^i \end{array}\right]\right\}_{i=1}^{n-1}$ are the sensor readings in the past, i.e. $\{t_i\}_{i=1}^{n-1} = \{t_1, \ldots, t_{n-1}\}$ which are the input features for the LSTM model and $W$ and $b$ are the matrix of weights and biases of the network, which get adjusted during the training process. Considering $y_1 = y_{st}^i - y_{ab}, y_s$, the maximum scour depth during the forecast window of $t$ can be written as:

$$y_s = \max (y_{st}^i - y_{so}^i)_{k=1}^n - y_1$$  \hspace{1cm} (2)

$$= y_{ab} - y_{so}^i$$  \hspace{1cm} (3)

$$= \chi\left(\left[\begin{array}{c} y_{st}^i \\ y_{so}^i \end{array}\right]\right)_{k=1}^{n-1}, W, b$$  \hspace{1cm} (4)

where $\chi$ represents the AI surrogate model approximating an scour empirical equation.

3. Results

3.1. LSTM performance comparison

A preliminary analysis was performed to better demonstrate the LSTM capability over a baseline model and a Dense model. The baseline model was defined as a model that at each time-step outputs the value of the previous time-step, i.e. for $t_i \in t$, $\left[\begin{array}{c} y_{st}^{i+1} \\ y_{so}^{i+1} \end{array}\right] = \left[\begin{array}{c} y_{st}^i \\ y_{so}^i \end{array}\right]$. The Dense model was a fully connected feed-forward neural network with one hidden layer (also called a Dense
layer) with ReLU activation function. The performance of the models was compared based on MAE over the test dataset on a smaller fraction of the data for bridge 742. Figure 13 shows that LSTM significantly outperforms the baseline and the Dense model, demonstrating LSTM superiority in forecast compared to the other models.

### 3.2. Model optimisation and hyper-Parameter tuning

Grid-search analyses were computationally-intensive; however, the computational time was significantly reduced by running the codes on the GPU clusters. Each training epoch took about 400-500 seconds on two GPU node. For each configuration, the LSTM network was retrained from scratch for a large number of times to evaluate the uncertainty in the training process. Figures 14 to 16 provide the box-plots of MAE values for various LSTM configurations. Note that the MAE values are based on normalised data (therefore unitless). Configurations were encoded as follows:

\[
\text{[features]} - \text{[model]} - \text{[window]} - \text{[units]} - \text{[drop-out]}
\]

For example, \texttt{ssy-ss-(336,168)-256-0} corresponds to the configuration that includes time features, uses the Single-Shot model, has 336 time-steps as input and 168 time-steps for labels, includes 256 units and does not have drop-out regularisation. The best configurations were selected based on the smallest MAE mean and standard deviation (see Table 1 for selected hyper-parameter values).

The following observations were made from these analyses:

- Adding time features did not improve the LSTM predictions of bed elevation, yet their presence showed to slightly improve the predictions of river stage for bridge 742.
- Increasing the input width from 336 hrs (14 days) to 720 hrs (30 days) while keeping the label width (forecast window) constant did not improve the prediction accuracy and showed to amplify the variability in model predictions.
- Increasing the label width (forecast window) from 7 days to 14 days showed to consistently increase the average prediction error.
- Increasing the LSTM layers to two didn’t show a meaningful improvement in the accuracy of predictions but increased the variability of predictions. This may be due to over-fitting, where the more complex model (more nonlinear units) learns better on the training data but fails to generalise on the test data.

![Figure 12. Schematic of flow and scour around a bridge pier, showing relevant parameters (Arneson et al. 2012).](image1)

![Figure 13. Comparison of Baseline, Dense and LSTM models (MAE is normalised).](image2)
The Feedback model did not outperform the Single-Shot model. This could be related to the error cascade effect from the initial steps within the label sequence to the next time steps. As the analyses have been performed on a limited number of bridges, the Feedback model may outperform the Single-Shot model in other bridges, depending on the training data.

Using the drop-out method (0.2 ratio) did not improve the accuracy of predictions.

The optimum number of units showed to vary from bridge to bridge and dependant on the other hyper-parameters of the model (e.g. feature combination, model variant, and architecture).

Figure 14. Grid-search results showing, (a) stage MAE and (b) sonar MAE boxplots for various configurations, with the top three models highlighted – bridge 742 (MAE is normalised).
3.3. Training

Figure 17 provides the graphs of the training process for a number of LSTM configurations, showing the variation of MAE in training and validation versus the number of epochs. As mentioned earlier, the stopping criteria was applied based on the error progression over

Figure 15. Grid-search results showing, (a) stage MAE and (b) sonar MAE boxplots for various configurations, with the top three models highlighted – bridge 539 (MAE is normalised).
validation dataset to avoid over-fitting. Note that the MAE is based on the standard-normalised values. Figure 18 provides a comparison between the two evaluation metrics, loss (MSE) and MAE across training epochs. Both metrics show consistency in their trends across epochs. The main difference between these two metrics is that, loss (MSE) is calculated based on the error on the vector of features, i.e. both sonar and stage predictions, whereas MAE is calculated separately for sonar and stage to better evaluate of scour prediction accuracy.
3.4. Prediction results

The AI forecast of river stage and bed elevation variation for the three case study bridges are presented and discussed in this section. The forecast results are shown for the test dataset, which is the last year of bridge monitoring data. The results pertain to the most optimum model selected through the grid-search analyses, as explained in the previous section.

3.4.1. Chilkat river bridge – 742

Figure 19(a) shows the data division for training, validation and test datasets. Figure 19(b) presents bed elevation and river stage predictions throughout 2017 (test dataset). The top graph shows the processed sensor readings and a shade created by the prediction of the time-steps for every data sequence over the label width. For bridge 742, the shade shows the predictions of 168 time-steps (7 days) into the future for every 336 time-steps of history (14 days), each of which is shifted to the right by one time-step. The shades closely follow the trends of the actual readings for both sonar and stage, showing the model’s effectiveness for scour forecast. On the bottom part, besides the actual readings, the average and 95% confidence interval of the predictions are shown. The mean absolute error of bed elevation (sonar) predictions over the test dataset is 0.19 m. Also the variability of predictions is within acceptable limits, with a standard deviation less than 0.3 m (1ft).

As mentioned earlier, bridge 742 shows a seasonal variation of bed elevation of up to 3 m, including about 2 m scour from June to August and subsequent 3 m of filling from August to October. The LSTM model shows to successfully follow the seasonal trend of scour and filling throughout 2017, having learned from long-term patterns in the training data from 2007 through 2016. During the scour episodes, the lower bound predictions consistently provide an accurate assessment of bed elevation variation and upper bound predictions closely track the bed elevation variations during filling episodes. Using the lower bound for scour and upper bound for filling episodes, the AI predictions show to provide an accurate conservative estimate of the bed elevation (hence scour assessment) at any given time.

3.4.2. Knik river bridge – 539

The data division for training is shown in Figure 20(a). The bed elevation and river stage predictions throughout 2016 (test dataset) is shown in Figure 20(b). Similar to bridge 742, the forecast is performed 7 days (168hrs) in advance with 14 days (336hrs) of history. The shades of LSTM predictions closely follow the trend of the actual sonar and stage readings. The mean absolute error of bed elevation prediction is about 0.25 m.
the test dataset. The variability of predictions is within acceptable limits with a standard deviation less than 0.3 m (1ft).

Similar to bridge 742, the AI predictions successfully follow the seasonal scour and filling trends throughout 2016, even during the acute changes.

Figure 19. (a) Data partitioning for AI training, (b) AI predictions, mean and confidence intervals over the test dataset- bridge 742.
between August to September with about 4 m variation of bed elevation. The variability in bed elevation predictions is small, with a maximum standard deviation below half a metre during the pick of filling in September 2016.

3.4.3. Sheridan glacier bridge – 230

The data division and AI predictions for bridge 230 are shown in Figure 21(a-b). More variability in bed elevation and river stage predictions is observed for bridge 230 compared to the other two bridges; nevertheless the trend (mean) of predictions follows the scour and filling episodes throughout 2017 (test dataset). This includes consecutive scour and filling episodes happening between August and September, with a maximum bed elevation variation of about 2 m (6.5ft). Similar to the other two bridges, the forecast is performed 7 days (168hrs) in advance with 14 days (336hrs) of history. The mean absolute error of bed elevation prediction is less than 0.37 m over the test dataset. It is worth noting that although the mean of prediction provides a reasonable forecast of scour at the bridge pier, the higher variability in predictions have resulted in the lower bound to indicate larger (more conservative) scour depth in some episodes, for e.g. around end of August and mid-September 2017.

3.5. Discussion on the performance of the LSTM models

Table 2 summarises the performance of the selected LSTM models for each bridge based on MAE for sonar and stage predictions. The average prediction error for bed elevation varies between 0.2 to 0.4 m and for stage water between 0.1 to 0.2 m. The average scour error percentage (= sonar MAE / max depth of scour) is less than 10% for bridge 742 and 539 but jumps higher to 25% for bridge 230. The reason for this larger error lies with a considerable amount of missing data (year 2008 and 2009) for bridge 230, as opposed to the other bridges. To provide a comparison basis with best of the empirical models, according to Sheppard, Melville, and Demir (2014), HEC-18 mean absolute error percentage over a substantially large lab dataset has been estimated around 21%.

The maximum error in scour trough and filling peak forecasts are provided in Table 3 and graphically shown in Figure 22. The maximum error based on the mean of predictions varies between 0.5 m to 0.7 m for scour troughs and 0.4 m to 1.7 m for filling peaks. The lower bound (LB) and upper bound (UB) errors show a reasonable degree of variability in the LSTM predictions, varying between 0.2 m to 0.9 m for scour and 0 m to 1.4 m for filling.

As mentioned in the previous sections, despite the clear seasonality in scour depth variation throughout the years, incorporating time (seasonality) features, i.e. [year sin] and [year cos] did not significantly improve the LSTM performance for 7-days forecast window. LSTMs are expected to capture both short and long-term temporal and seasonal patterns therefore, this outcome is not surprising; however, the impact of seasonality features on the prediction accuracy is expected to increase for longer forecast windows. This topic will be studied in future phases of this research.

3.6. Impact of flow velocity (Discharge)

In order to explore whether velocity is a critical feature in presence of stage time-series, we incorporated the discharge measurements (discharge), obtained from the USGS website, into the LSTM models for bridge 742 as an input feature and compared the performance among three different feature combinations: ssd:[sonar, stage, discharge], sd:[sonar, discharge] and ss:[sonar, stage]. The USGS measurements at gaging stations are fully explained in Sauer and Turnipseed (2010). The most practical method of measuring the discharge of a stream is the velocity-area method. Discharge is computed as the product of the area and velocity. The measurement is made by subdividing a channel cross section into segments and by measuring the depth and velocity in a vertical within each segment. The velocity is measured through various methods such as current-meter, ADV (Acoustic Doppler Velocimeter) and ADCP (Acoustic Doppler Current Profiler).

Figure 23 provides histograms of the discharge time-series for bridge 742 and its cross-correlation with sonar and stage. Also, the normalised discharge time-series data in 2017 together with sonar and stage are presented for bridge 742 in Figure 24. Stage and discharge show high positive correlation (nonlinear) as observed both in Figures 23 and 24.

As shown in Figure 25, the LSTM performance is not improved by adding discharge as an additional input feature. Replacing stage with discharge did not contribute to better performance either. This is due to the high correlation between stage and discharge and confirms that stage and sonar input time-series are sufficient for LSTM scour forecast models.
Figure 20. (a) Data partitioning for AI training, (b) AI predictions, mean and confidence intervals over the test dataset – bridge 539.
Figure 21. (a) Data partitioning for AI training, (b) AI predictions, mean and confidence intervals over the test dataset – bridge 230.
4. Conclusions

This study introduces a novel AI approach for the real-time forecast of scour around bridge piers, as a basis for an AI-based early warning system using scour monitoring sensors. Many scour-critical bridges across the US have been monitored since early 2000s using stage and sonar sensors, as well as frequent soundings and surveys. A collaborative research effort with USGS and DOTs enabled access to some of this data. The data from a number of these bridges in Alaska was used in this study to train AI algorithms to predict both river stage (flood/water level) and bed elevation variations as early as seven days in advance.

| Bridge | Sonar MAE-m | Stage MAE-m | Max scour depth-m | Scour error-% |
|--------|-------------|-------------|-------------------|---------------|
| 742    | 0.19        | 0.19        | 2.1               | 9             |
| 539    | 0.25        | 0.13        | 3.3               | 7.5           |
| 230    | 0.37        | 0.08        | 1.5               | 25            |

Table 2. Performance summary of the LSTM models over test dataset.

| Bridge | Mean error (m) | LB error (m) | Mean error (m) | UB error (m) |
|--------|----------------|--------------|----------------|--------------|
| 742    | 0.7            | 0.2          | 0.4            | 0            |
| 539    | 0.6            | 0.6          | 1.7            | 0.8          |
| 230    | 0.5            | 0.9          | 0.4            | 1.4          |

Table 3. Maximum prediction error for scour (troughs) and filling (peaks) across the test dataset.

Figure 22. Definition of maximum errors based on mean and LB/UB predictions for scour troughs and filling peaks: (a) bridge 742, (b) bridge 539, (c) bridge 230.
The proposed solution was validated using several bridges in Alaska and case studies were shown for three of these bridges with the most significant scour and flood events.

The LSTM networks were trained using historical monitoring data including eleven years of sonar and stage (bed elevation and river stage) time-series and were shown to provide reliable early forecasts of scour with reasonable uncertainty. Results showed that LSTMs can learn the long-term seasonal patterns in both stage and sonar data and make reasonably accurate predictions for upcoming changes. The mean of the LSTM predictions was shown to capture the trend of variations of bed elevation during seasonal cycles of scour and filling, happening between July to October each year in Alaska. Also, the lower and upper bound

Figure 23. Histograms of discharge data – bridge 742.

Figure 24. Sample of discharge, stage and sonar time-series – bridge 742 (normalised data).

Figure 25. Comparing the LSTM model performance with different feature combinations including discharge for, left) validation dataset, right) test dataset – bridge 742.
of bed elevation predictions provided reliable, conservative estimates of scour and filling depth during the abrupt changes due to high flows and flood events.

Discharge time-series data showed to be highly correlated with stage and adding that as an input feature did not improve the LSTM performance. Also, the other relevant parameters in typical scour empirical models, such as riverbed/soil condition and pier geometry are not temporally variable, therefore not considered as input features for the LSTM, allowing for low-dimensional models.

Among more than 300 different LSTM configurations evaluated, the optimum models yield an average scour prediction error of 0.2 to 0.35 m. In comparison with the empirical models, the proposed LSTM models offer the following advantages for scour forecast:

- The LSTM scour forecast model is able to provide real-time prediction of upcoming scour, within a pre-defined forecast window.
- The average error of scour prediction using the LSTM model is 7.5–25%, showing a competitive performance against the empirical models with an error range of 7–170% (Sheppard, Melville, and Demir 2014).
- Given that each bridge has its particular trends of scour and flooding depending on the location, structure, river flow and river bed material and condition, training the LSTM scour forecast model should be performed individually for each bridge. Therefore, the customised LSTM model is inherently aware of the trend of scour particular to the bridge pier location, through learning from historic flow and bed elevation variation patterns.
- Unlike the empirical models which involve several parameters, the LSTM model is only dependant of two main time-series features, i.e. river bed and stage elevations, eliminating the need for measurements of other parameters, such as velocity and particle size.

The cost of implementing the proposed AI-based early warning solution for bridges with already installed stage and sonar sensors is negligible compared to potential failure costs, especially for critical bridges subjected to frequent and/or extreme flooding. It is expected that transportation authorities can benefit from the proposed solution to enhance their risk management systems for scour-critical bridges.

Acknowledgments

The authors also would like to thank Tony Marshall, Matt Carter and Dr Will Cavendish from Arup and Dr Farbod Taymouri from The University of Melbourne, for their valuable advice and support throughout the project.

Data Availability Statement

Some or all data used for this study are available from the corresponding author upon request. Models, or codes generated or used during the study are proprietary or confidential in nature and may only be provided with restrictions.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This study was funded by Arup Global Research Excellence Award (2020–2021).

ORCID

Negin Yousefpour 🌐 http://orcid.org/0000-0002-6634-2658

References

Arneson L., L. Zevenbergen, P. Lagasse, and P. Clopper. 2012. “Evaluating Scour at Bridges-Fifth Edition, Hydraulic Engineering Circular No. 18.” Federal Highway Administration (FHWA).

Azamathulla H. M., A. A. Ghani, N. A. Zakaria, and A. Guven. 2010. “Genetic Programming to Predict Bridge Pier Scour.” Journal of Hydraulic Engineering 136 (3): 165–169. https://doi.org/10.1061/(ASCE)HY.1943-7900.0000133.

Balestriero R., J. Pesenti, and Y. LeCun. 2021. “Learning in High Dimension Always Amounts to Extrapolation.” CoRR, abs/2110.09485.

Bengio Y., P. Simard, and P. Frasconi. 1994. “Learning Long-term Dependencies with Gradient Descent is Difficult.” IEEE Transactions on Neural Networks 5 (2): 157–166. https://doi.org/10.1109/TNN.72.

Briaud J.-L., S. Hurlebaus, K. Chang, C. Yao, H. Sharma, O-Y. Yu, and G. Price. 2011. “Real-Time Monitoring of Bridge Scour Using Remote Monitoring Technology.” Rep. No. 0-6060-1, Texas A&M University/TTI.

Chang W.-Y., J.-S. Lai, T.-Y. Yu, F. Lin, L.-C. Lee, W.-F. Tsai, and C.-H. Loh. 2014. “Pier Scour Monitoring System by Bed-level Image Tracking.” International Journal of Sediment Research 29 (2): 269–277. https://doi.org/10.1016/S1001-6279(14)60042-1.

Deng L., and D. Yu. 2014. “Deep Learning: Methods and Applications.” Report no., Microsoft (May).

Dikanski H., A. Hagen-Zanker, B. Imam, and K. Avery. 2016. “Climate Change Impacts on Railway Structures: Bridge Scour.” In Proceedings of the Institution of Civil Engineers (ICE) – Engineering Sustainability, 170.

Ebtehaj I., A. M. A. Sattar, H. Bonakdari, and A. H. Zaji. 2016. “Prediction of Scour Depth Around Bridge Piers Using Self-adaptive Extreme Learning Machine.” Journal of...
Yousefpour N., and M. Pouragha. 2022. “Prediction of the Post-failure Behavior of Rocks: Combining Artificial Intelligence and Acoustic Emission Sensing.” *International Journal for Numerical and Analytical Methods in Geomechanics* 46 (10): 1874–1894. https://doi.org/10.1002/nag.v46.10.

Yousefzadeh R., and X. Cao. 2022. “To What Extent Should We Trust AI Models When They Extrapolate?” *CoRR*, abs/2201.11260.

Zhang N., S.-L. Shen, A. Zhou, and Y.-F. Jin. 2021. “Application of Lstm Approach for Modelling Stress–strain Behaviour of Soil.” *Applied Soft Computing* 100:106959. https://doi.org/10.1016/j.asoc.2020.106959.

Zhang P., Z.-Y. Yin, Y.-F. Jin, and G.-L. Ye. 2020. “An AI-based Model for Describing Cyclic Characteristics of Granular Materials.” *International Journal for Numerical and Analytical Methods in Geomechanics* 44 (9): 1315–1335. https://doi.org/10.1002/nag.v44.9.

Zounemat-Kermani M., A.-A. Beheshti, B. Ataie-Ashtiani, and S.-R. Sabbagh-Yazdi. 2009. “Estimation of Current-induced Scour Depth Around Pile Groups Using Neural Network and Adaptive Neuro-fuzzy Inference System.” *Applied Soft Computing* 9 (2): 746–755. https://doi.org/10.1016/j.asoc.2008.09.006.