A Long-term Probabilistic Forecasting Approach of TBM Operating Parameters based on Deep Learning

Speaker: Jinpu Cao 4-5 August 2022

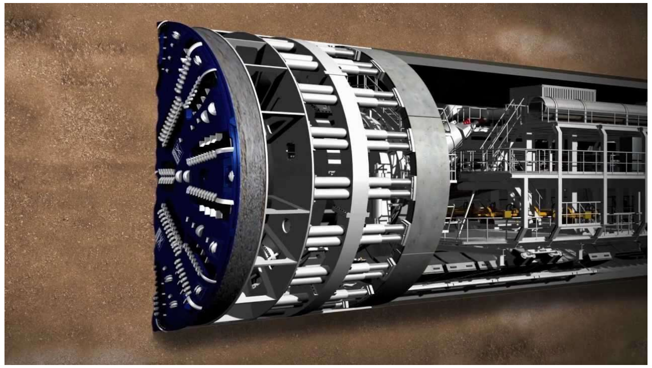
4th International Conference on Information Technology in Geo-Engineering

Problem & Motivation

Dilemmas & uncertainties:

- When a Tunnel Boring Machine (TBM) is working, there will always be some unexpected poor surrounding geological conditions, such as fault, rock burst, groundwater inrush …

- The TBM will easily break down if the equipment’s control parameters do not adapt to these special conditions in time.



Tunnel Boring Machine

rock burst

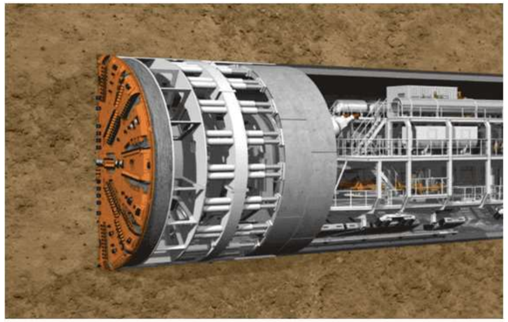
groundwater inrush

Problem & Motivation

Dilemmas & uncertainties:

- unexpected poor surrounding geological conditions
- TBM parameters’ adaptation to the special conditions in time

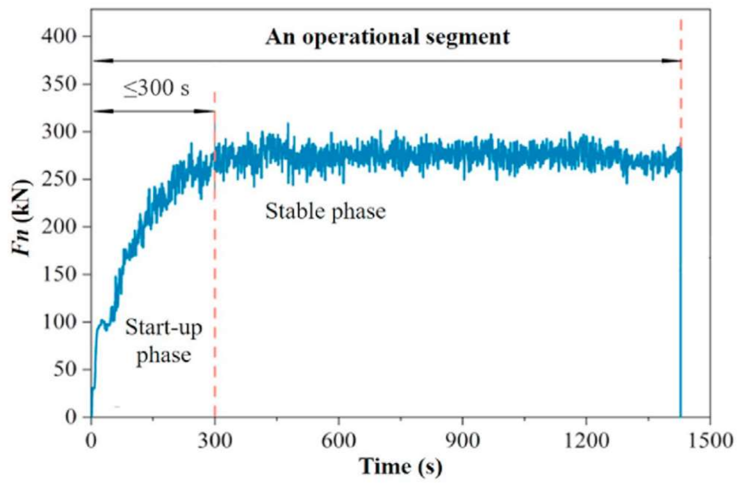
To address these, we want a model to tell us, under normal conditions, how the operating parameters, such as thrust and torque, should change in the long-term future.



Forecast the thrust for the next *one minute* during the TBM tunneling

Related Work

- In a tunneling cycle, given the data in the start-up phase, the prediction of operating parameters in the stable phase is one of the most important topics and has been well explored (Wang, Xin, et al. Chen, Haowen, et al.)
- The real-time parameter prediction has also been explored (Gao, Xianjie, et al.). However, the prediction time is too short to be meaningful enough (Erharter, Georg H., and Thomas Marcher.).
- Little research studies TBM parameters probability prediction, which is more useful in practice.



Partition of the start-up and stable tunneling phases (Huang, Xing, et al.)

Methodology

A long-term probabilistic forecasting approach of TBM operating parameters based on deep learning：

1. Define and instantiate the forecasting problem

2. Prepare and augment TBM tunneling data

   Data description, data preparation, feature selection, data augmentation

3. Conduct probabilistic forecasting with autoregressive recurrent networks

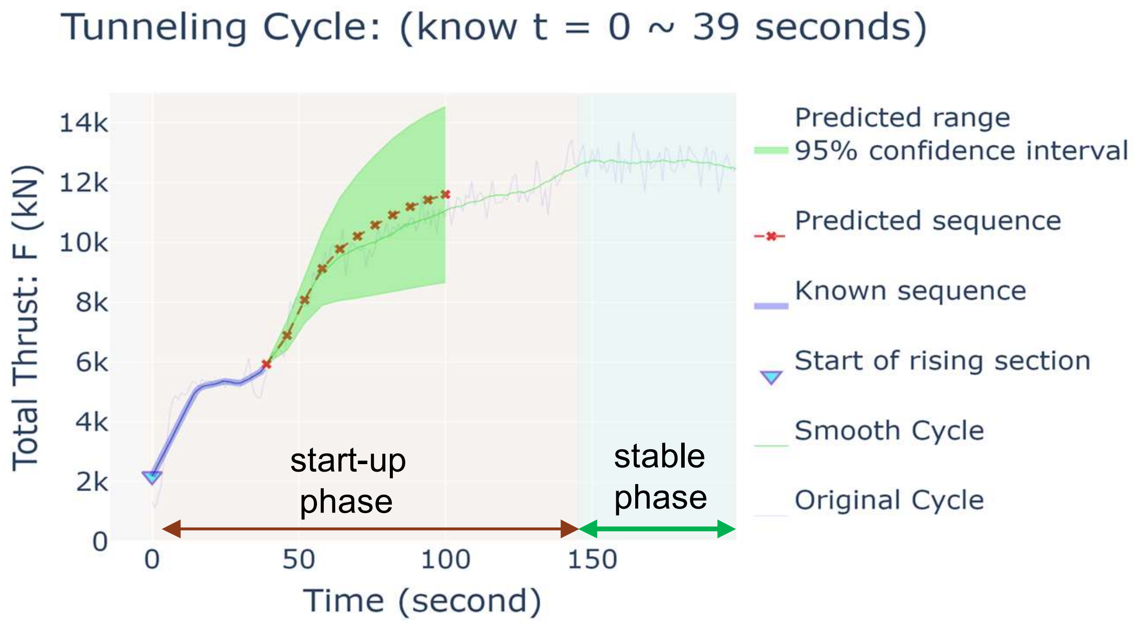
   DeepAR networks, model architecture, metrics

Methodology

1. Define and instantiate the forecasting problem

- Problem Definition

\- Given 40-second TBM data in the start-up phase, forecast the thrust and torque sequence with a certain confidence interval in the **next 1 minute** (represented by 10 equally spaced points)



Methodology

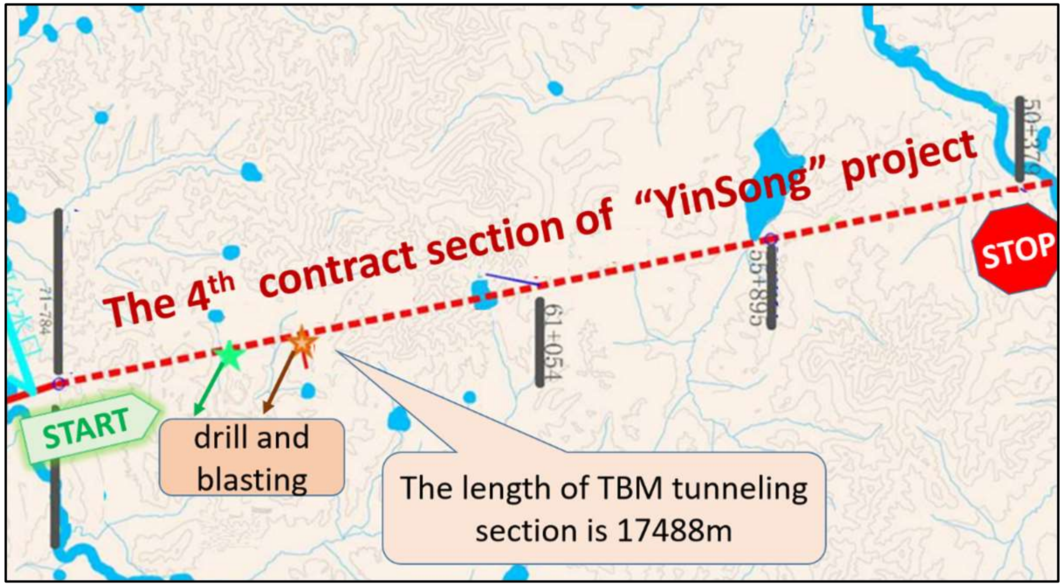
2. Prepare and augment TBM tunneling data

- Data description

- Data collected by **CREG** in the 4th contract section of **Yinsong Water Diversion Project, China.**

- **198** TBM operational parameters

- Sampling at every second covering **728 days**



Methodology

2. Prepare and augment TBM tunneling data

- Data Preparation

- Extract tunneling cycles based on key parameters, including velocity, rotation rate, thrust and torque, are non-zero. (**8214** cycles)

- Remove outliers based on 3-σ criterion.

- For each cycle, identify the beginning time of the start-up phase according to the friction and idling torque that must be overcome in the process of tunneling.
  Thrust: 4480 kN Torque: 224kN·m (Li, Jianbin, et al.)

Methodology

2. Prepare and augment TBM tunneling data

- Feature Selection

- Using the Pearson correlation coefficient, **6 tunneling parameters** closely related to thrust and torque and relatively independent of each other are selected, including velocity, rotation rate, thrust, torque, power, and penetration (Lv, Jiajun)

- Smooth the data with moving average (size = 20 seconds)

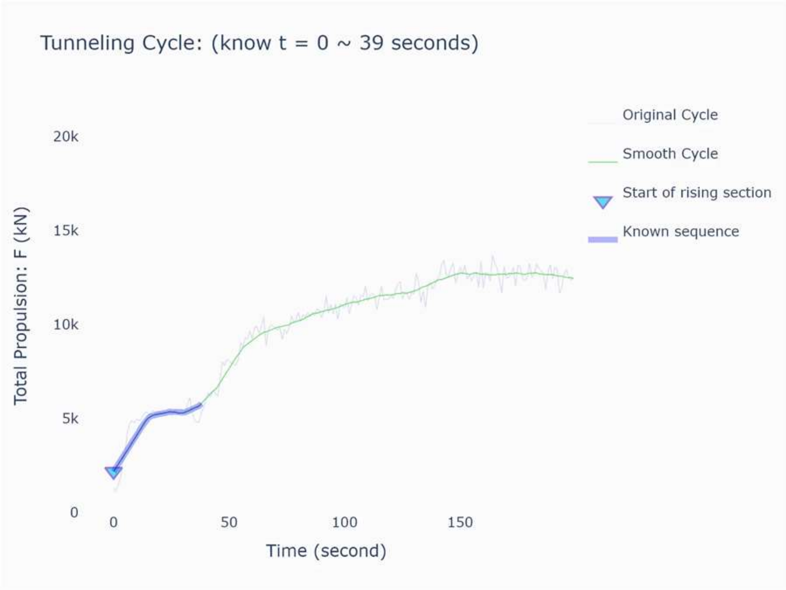
- To eliminate dimensional differences and speed the training process, normalize the data to 0 ~1

Methodology

2. Prepare and augment TBM tunneling data

- Data augmentation

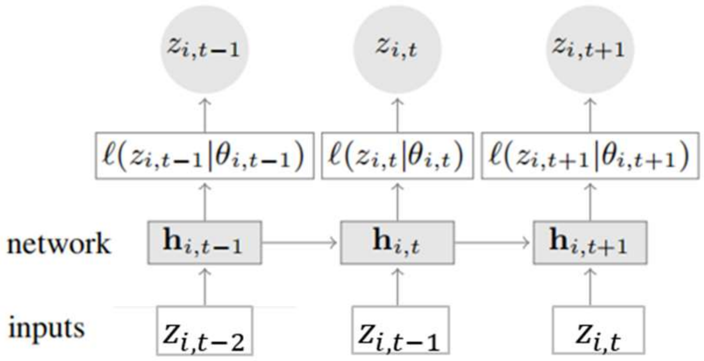
- Extract more than one sample in one tunneling cycle.
  e.g., 1st sample 1~40 seconds → next 1 minute
  2nd sample 2~41 seconds → next 1 minute



Methodology

3. Conduct probabilistic forecasting with autoregressive recurrent networks

- DeepAR is a supervised learning algorithm for time series forecasting that uses recurrent neural networks (RNN) to produce both point and probabilistic forecasts



- Model summary
  i: sample index t: time index
  z: time series
  h: network parameters
  θ: probability distribution parameter
  $l(z|\theta)$: likelihood function

- Model optimization

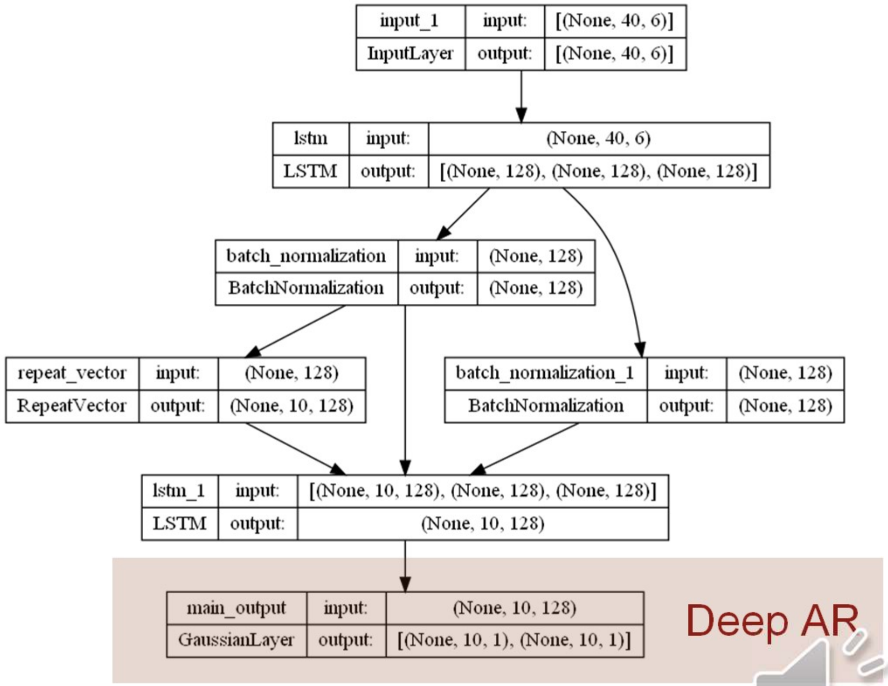
$$\text{maximize} \sum_{i=1}^{N} \sum_{t=t_0}^{T} l(z_{i,t}|\theta)$$

Methodology

3. Conduct probabilistic forecasting with autoregressive recurrent networks

- Model Architecture
- Hyperparameter
  2 layers of LSTM
  128 units of each layer
  learning rate: 0.001
- Assume Gaussian Distribution (Gaussian Layer)

$$\ell_G(z|\mu,\sigma) = (2\pi\sigma^2)^{-\frac{1}{2}} \exp(-(z-\mu)^2/(2\sigma^2))$$



Methodology

3. Conduct probabilistic forecasting with autoregressive recurrent networks

- Metrics
- Accuracy

$$Accuracy = \left(1 - \frac{1}{n}\sum_{i=1}^{n}\frac{1}{T^{(i)}}\sum_{t=1}^{T^{(i)}}\left|\frac{y_t^{(i)} - \hat{y}_t^{(i)}}{y_t^{(i)}}\right|\right) \times 100\%$$

- Mean Absolute Error (MAE)

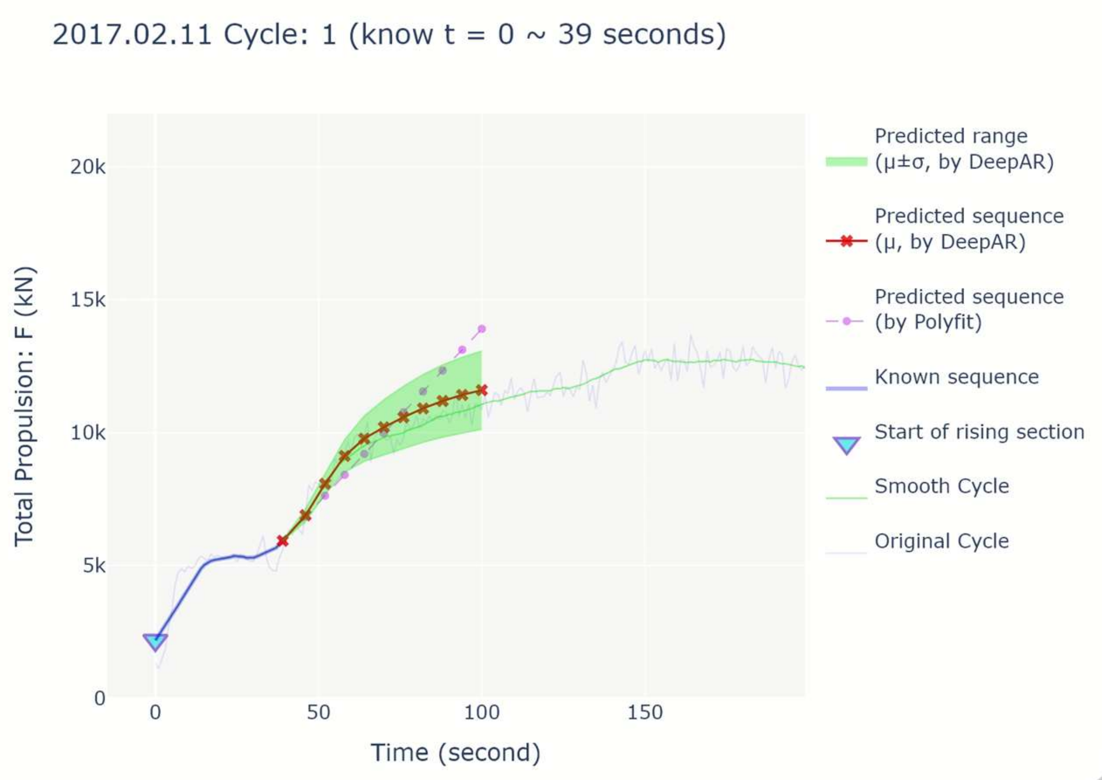
$$MAE = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{T^{(i)}}\sum_{t=1}^{T^{(i)}}\left|y_t^{(i)} - \hat{y}_t^{(i)}\right|$$

Results

Thrust Forecasting:

		DeepAR model	Polynomial fitting model
Accuracy (%)		93.7	84.0
MAE (kN)		657	1854
prediction within	μ±σ (68%)	71.7	-
	μ±2σ (95%)	94.6	-
	μ±3σ (99.7%)	98.7	-

Polynomial fitting model:
Use past two points to fit a line and forecast



Results

Torque Forecasting:

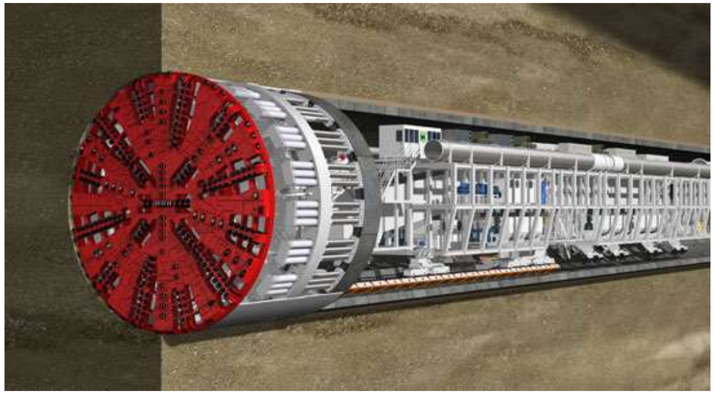
		DeepAR model	Polynomial fitting model
Accuracy (%)		82	72.46
MAE (kN·m)		202	393
prediction within	μ±σ (68%)	63.8	-
	μ±2σ (95%)	87.8	-
	μ±3σ (99.7%)	95.4	-

Polynomial fitting model:
Use past three points to fit a line and forecast

Conclusion & Future work

- Conclusion
  - The DeepAR-based probabilistic forecasting approach performs well when forecasting long-term TBM operating parameters in real-time. The accuracies of thrust and torque are up to 94% and 82%, respectively.

- Future work
  - Increase the accuracy of torque forecasting by more refined data processing
  - Explore the feasibility of the model in soft rock



Thanks for your attention

jinpu@stanford.edu