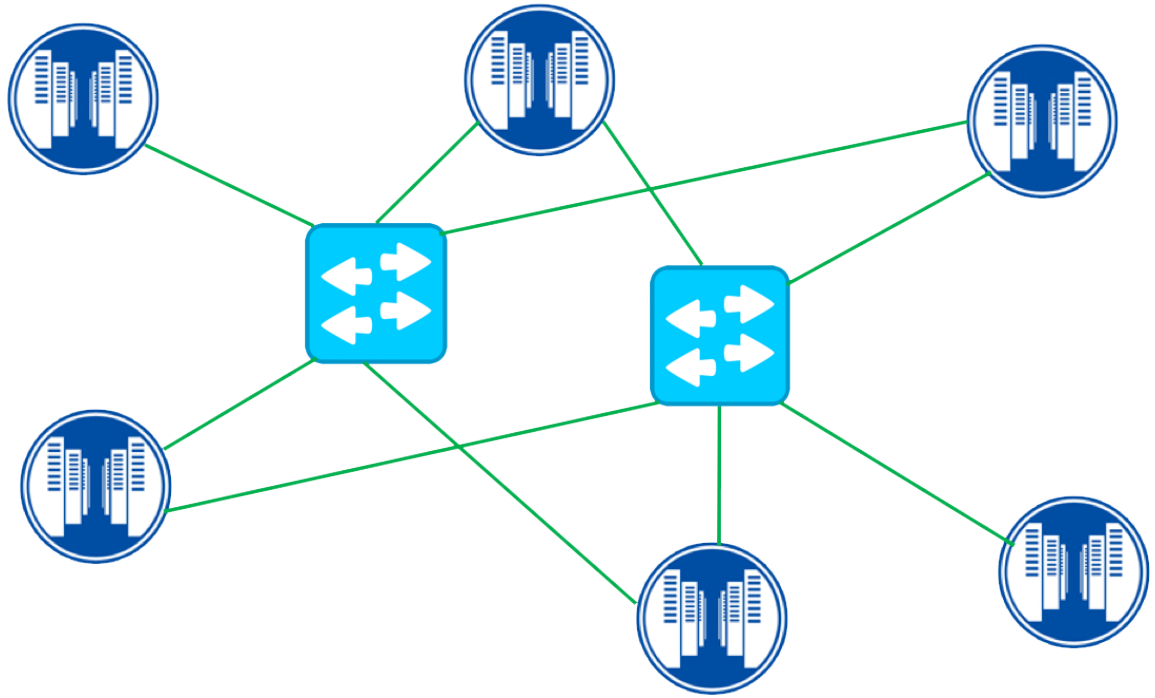
Inter-Data-Center Network Traffic Prediction with Elephant Flows

Yi Li*, Hong Liu†, Wenjun Yang†, Dianming Hu†, Wei Xu*
* Institute for Interdisciplinary Information Sciences, Tsinghua University
† Baidu Inc.

Inter-Data-Center Network Traffic



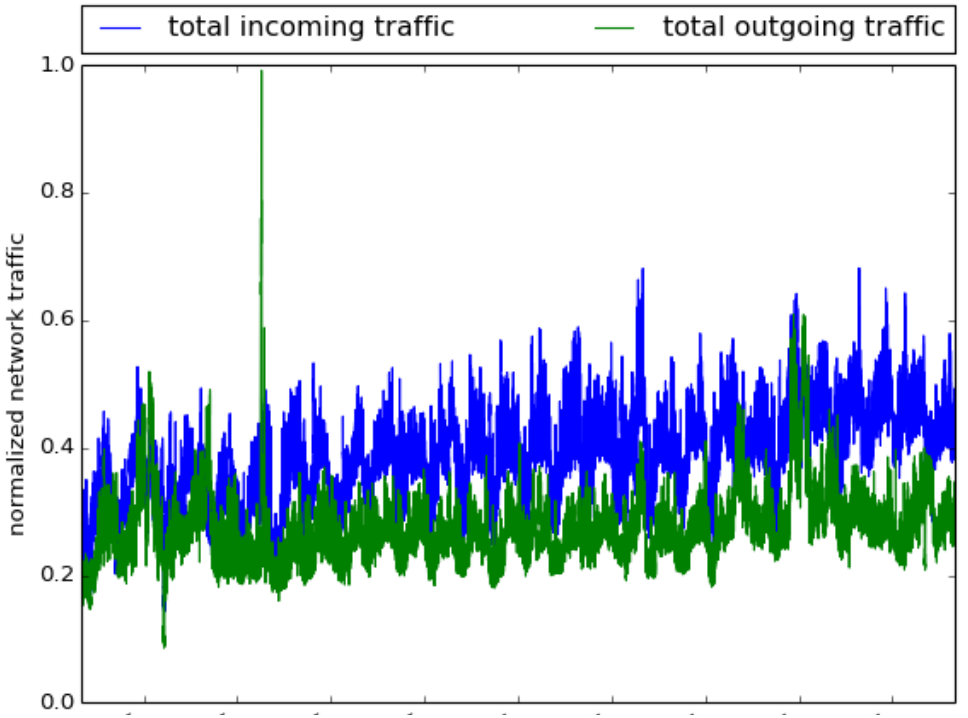
- Inter-DC traffic is growing with applications
  - Video streaming
  - File sharing
- Heavy inter-DC traffic with spikes and fluctuations
  - Congestions
  - Cost: ISPs charge by peak bandwidth
- Accurate inter-DC traffic prediction is important
  - Network resource provisioning
  - Traffic engineering.

Challenges in inter-DC traffic prediction

- **Neither representing linear processes nor having stable statistical properties.**
  - linear models, e.g. ARIMA, do not work well

- **Different patterns from Internet traffic**
  - bursty and unpredictable
  - conventional prediction methods for Internet traffic do not work well

- **No obvious recurring traffic patterns**
  - a small number of **elephant flows** dominates



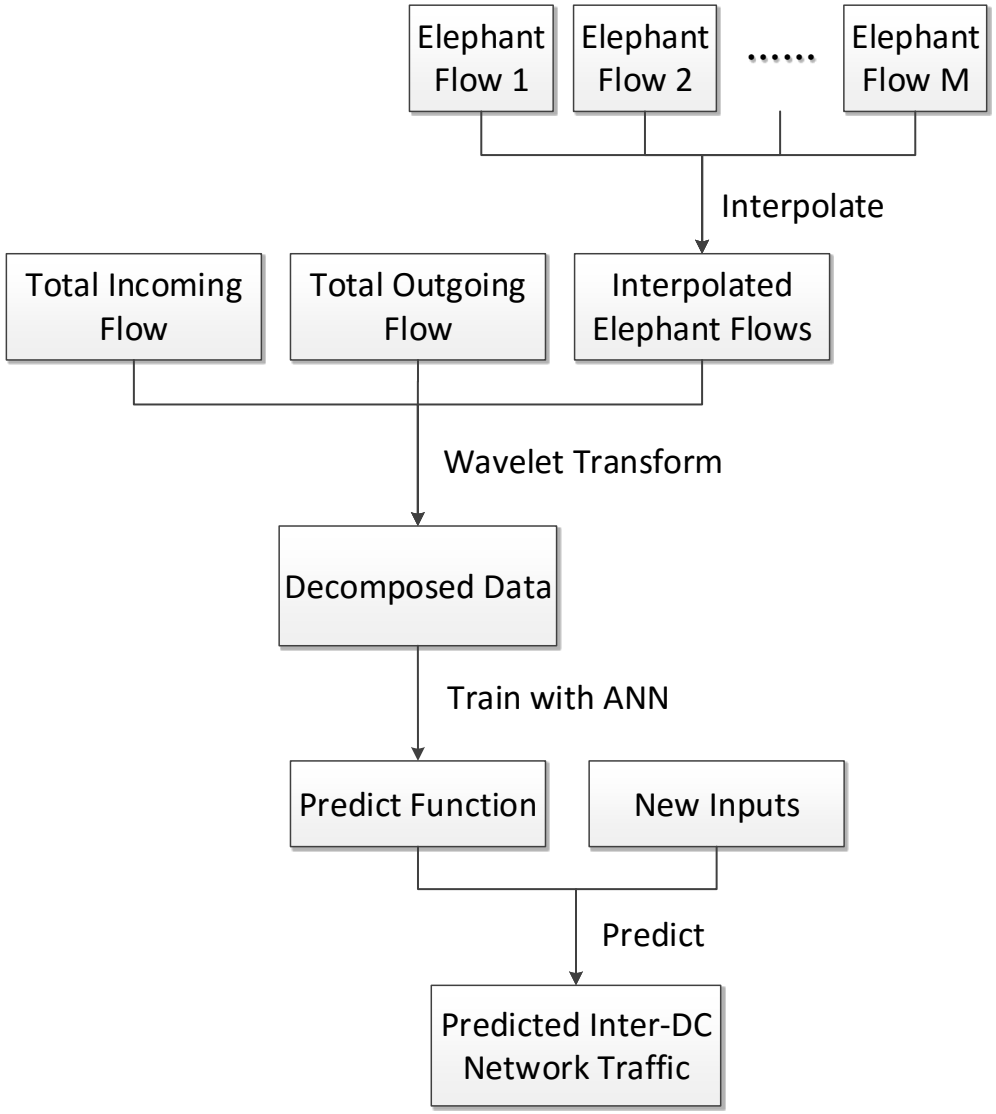
Our Contributions

- We propose a network traffic prediction model for inter-DC traffic by **treating elephant flows explicitly**.
- We introduce **effective interpolation method** to reduce the amount of expensive flow-level observations for the elephant flows.
- We evaluate our model on a real-world datacenter and help Baidu **reduce the peak bandwidth for about 9% on average**.

Model Overview

- Elephant flows
- Interpolation
- DB4 wavelet transform
- ANN
- RRMSE (Relative Root Mean Squared Error)

$$RRMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\frac{\hat{\theta}_i - \theta_i}{\theta_i})^2}$$



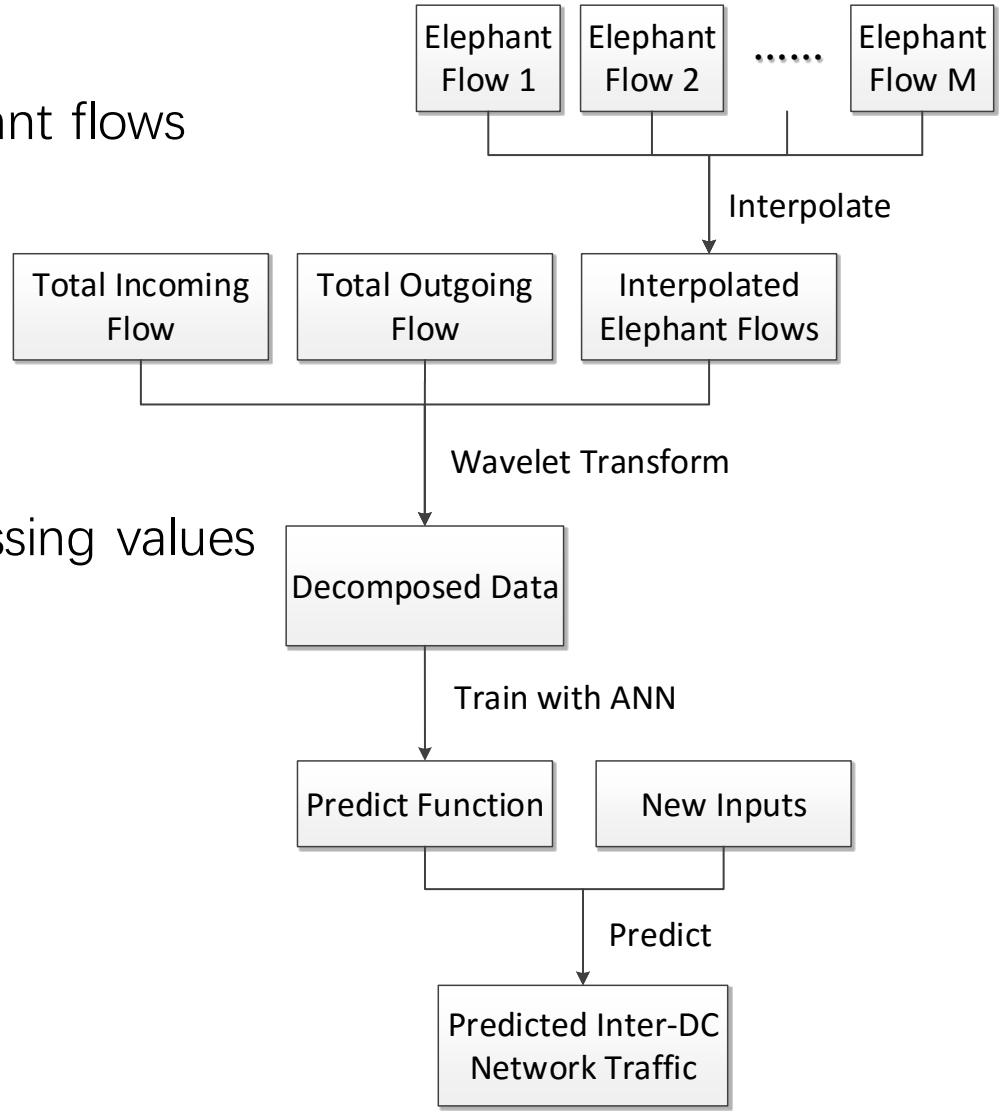
Model Overview

Total traffic

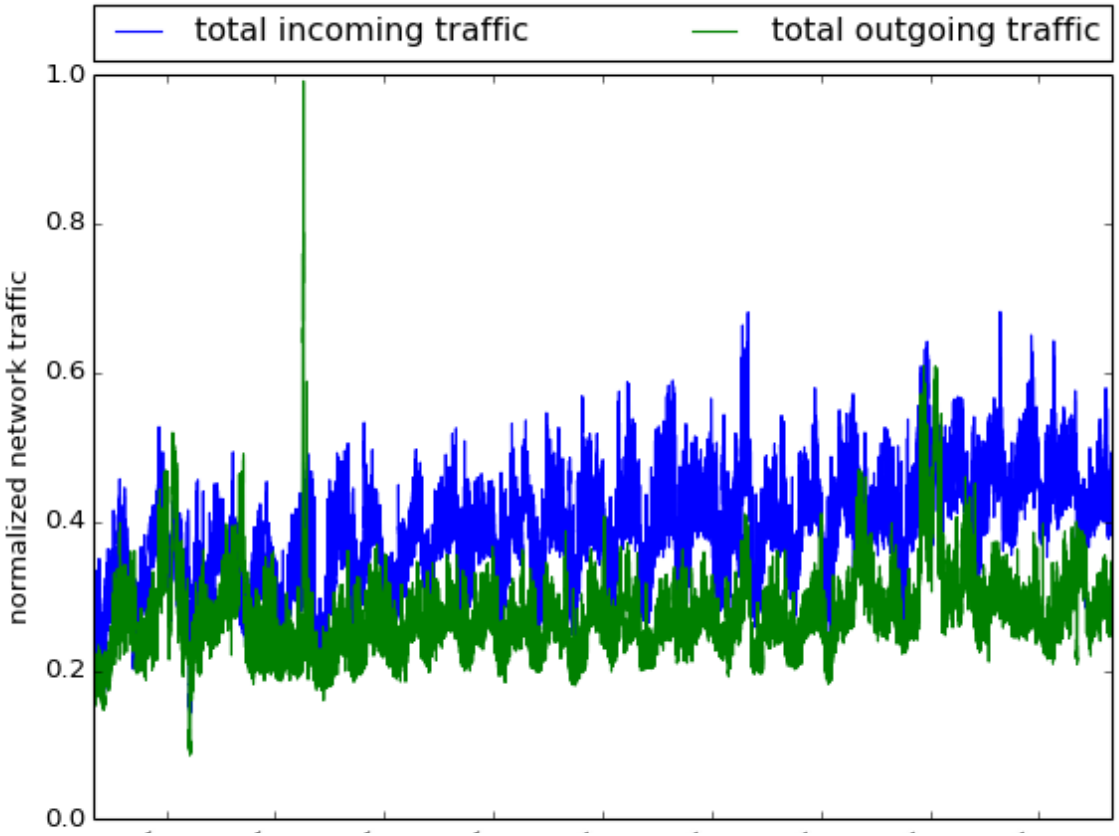
Traffic of elephant flows

in^{t+0}	out^{t+0}	ein_1^{t+0}	$eout_1^{t+0}$	…	ein_M^{t+0}	$eout_M^{t+0}$	
in^{t+1}	out^{t+1}	ein_1^{t+1}	$eout_1^{t+1}$	…	ein_M^{t+1}	$eout_M^{t+1}$	
in^{t+2}	out^{t+2}	ein_1^{t+2}	$eout_1^{t+2}$	…	ein_M^{t+2}	$eout_M^{t+2}$	
⋮							
in^{t+s}	out^{t+s}	ein_1^{t+s}	$eout_1^{t+s}$	…	ein_M^{t+s}	$eout_M^{t+s}$	
⋮							
in^{t+k}	out^{t+k}	ein_1^{t+k}	$eout_1^{t+k}$	…	ein_M^{t+k}	$eout_M^{t+k}$	

Missing values



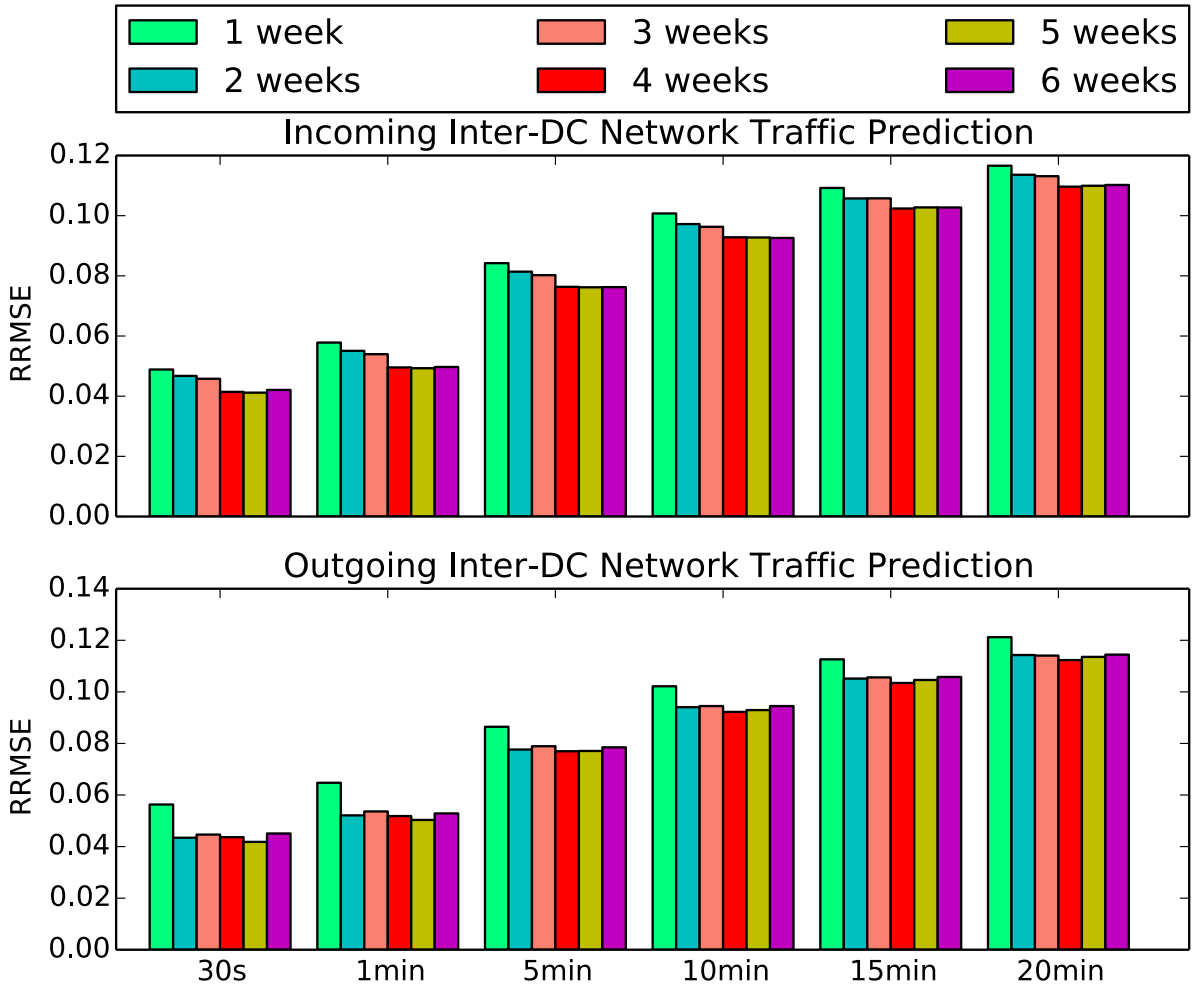
Data Collection



- Real traffic from a production data center in Baidu
- Counters on the data center edge routers using SNMP, every 30 sec.

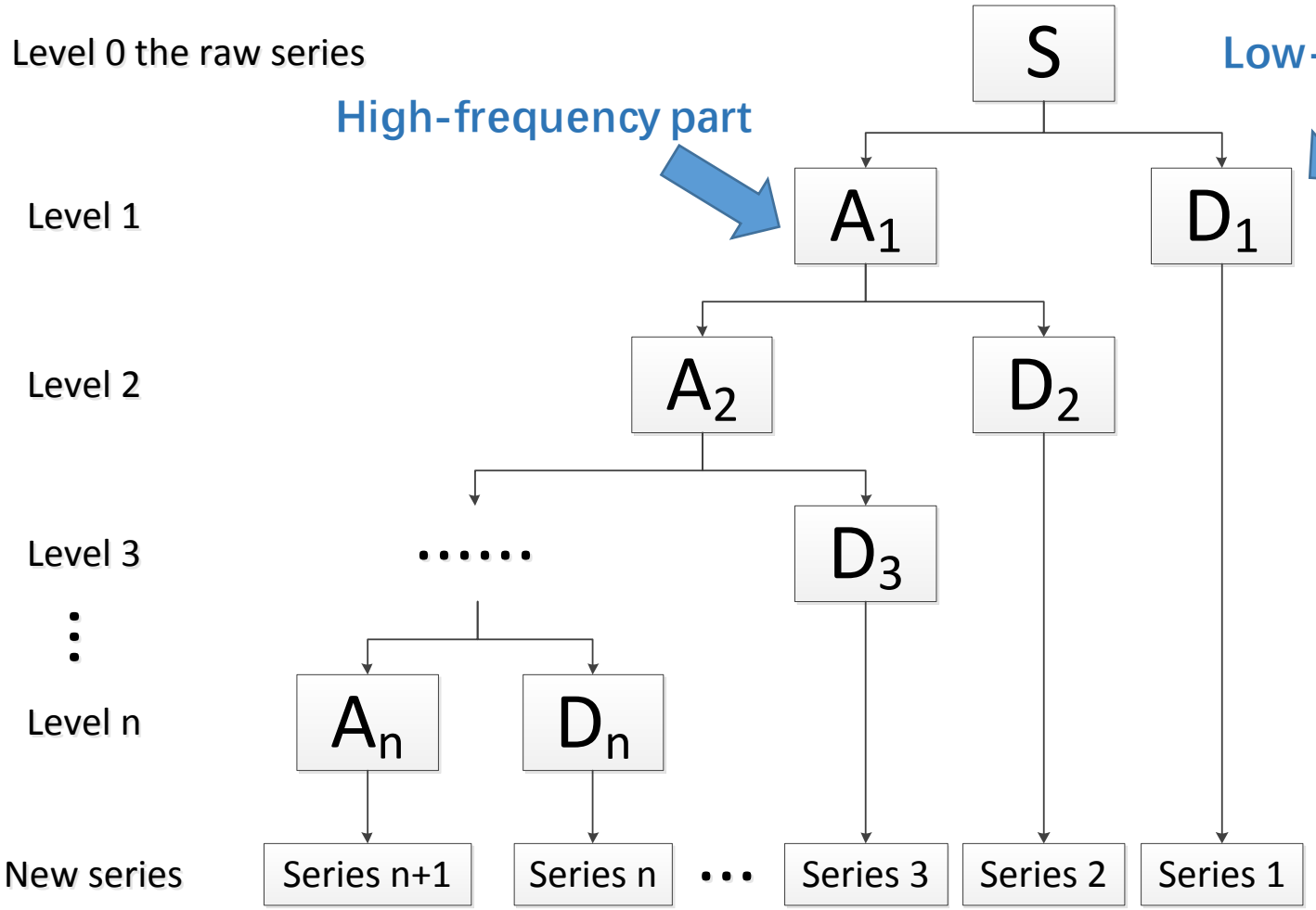
- Data of six weeks
- The data of the last day for testing

Length of Training Set



Training set of 4 weeks performs better!

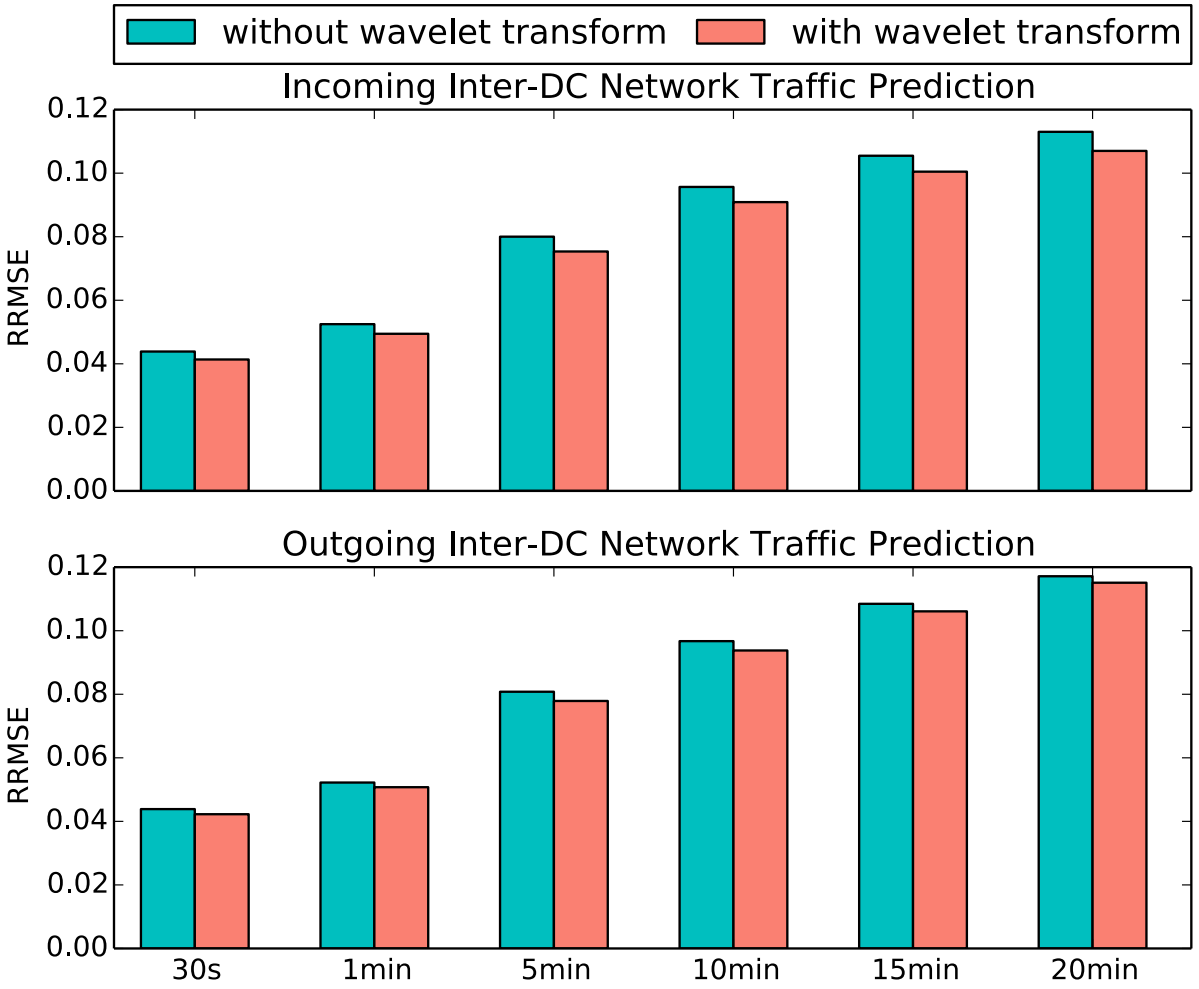
Wavelet Transform



$$X(a,b) = \frac{1}{\sqrt{b}} \int_{-\infty}^{+\infty} x(t)\Psi(\frac{t-a}{b})dt$$

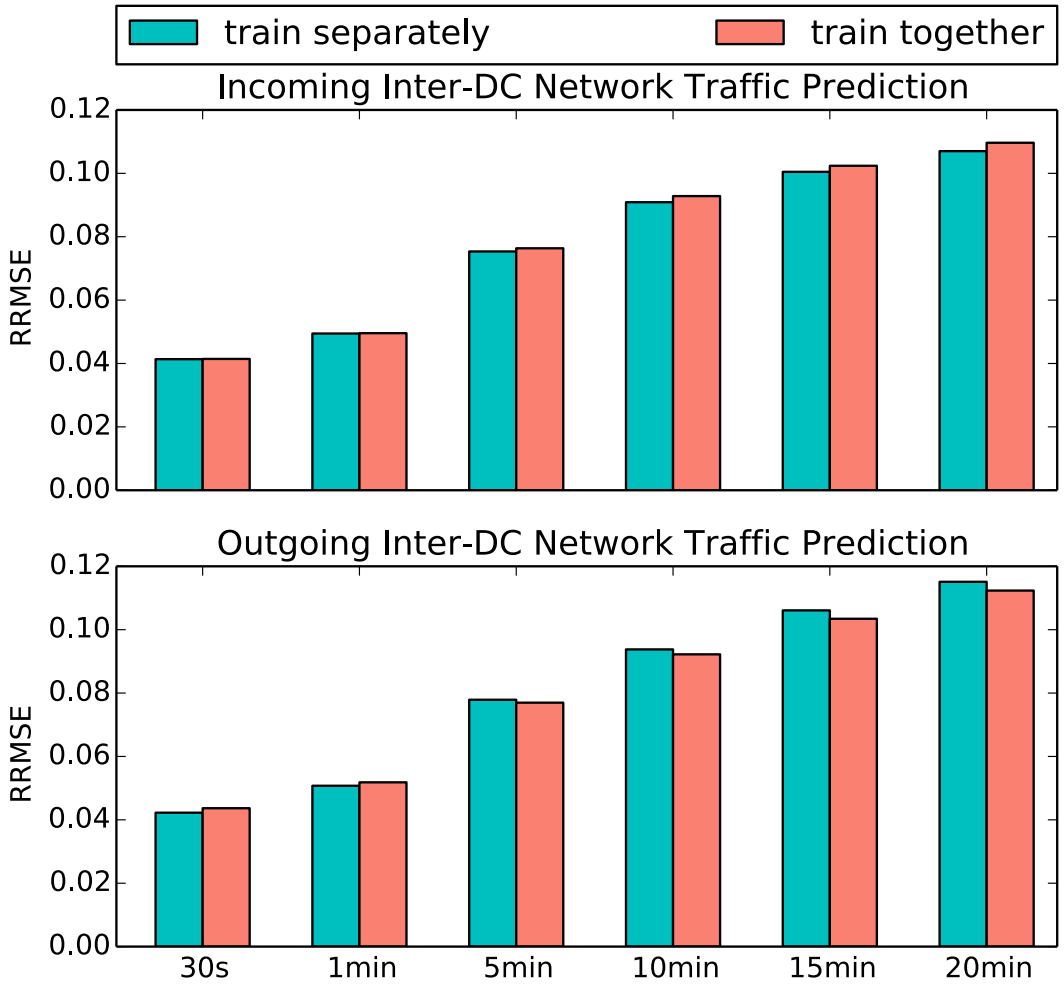
- DB4 wavelet transform
- 10 levels
- Each series is decomposed to 11 new series

Wavelet Transform



Training with wavelet transform performs better!

Combing Incoming/Outgoing Traffic



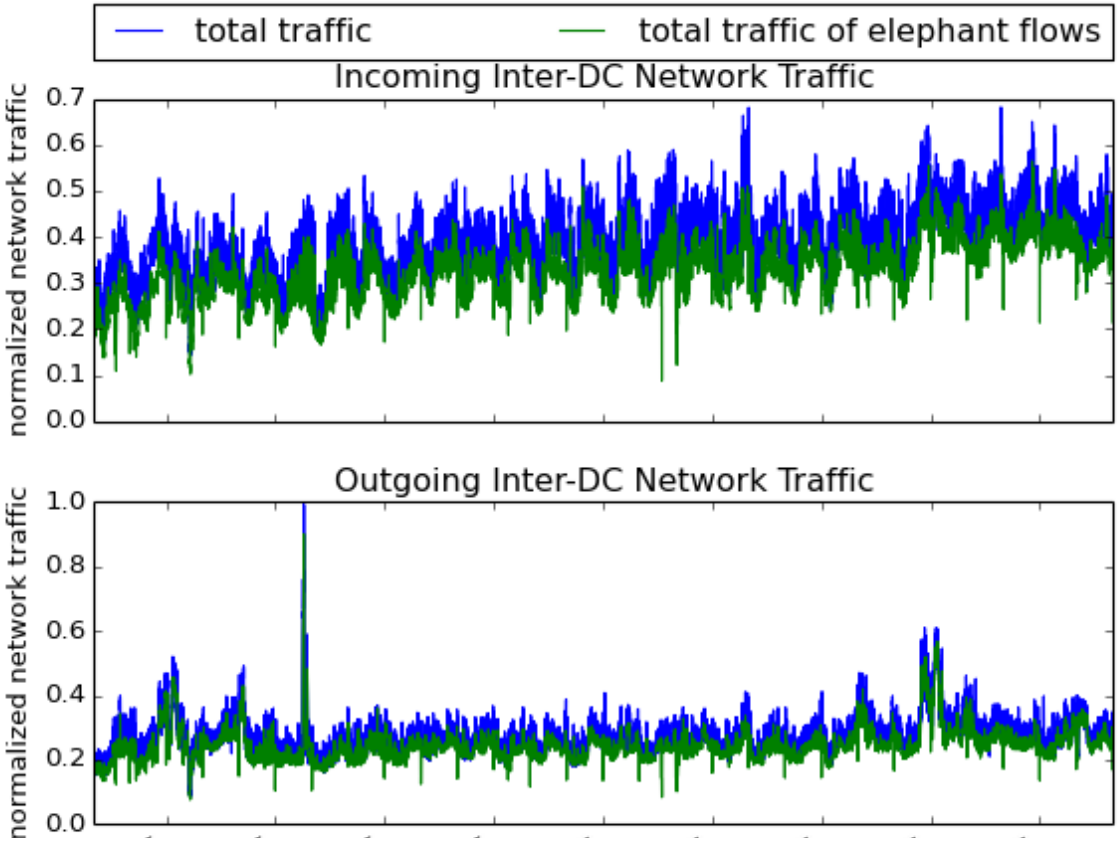
Combine incoming/outgoing traffic numbers into the same model

No significant difference in accuracy

But…

40% reduction in training time

Elephant flows

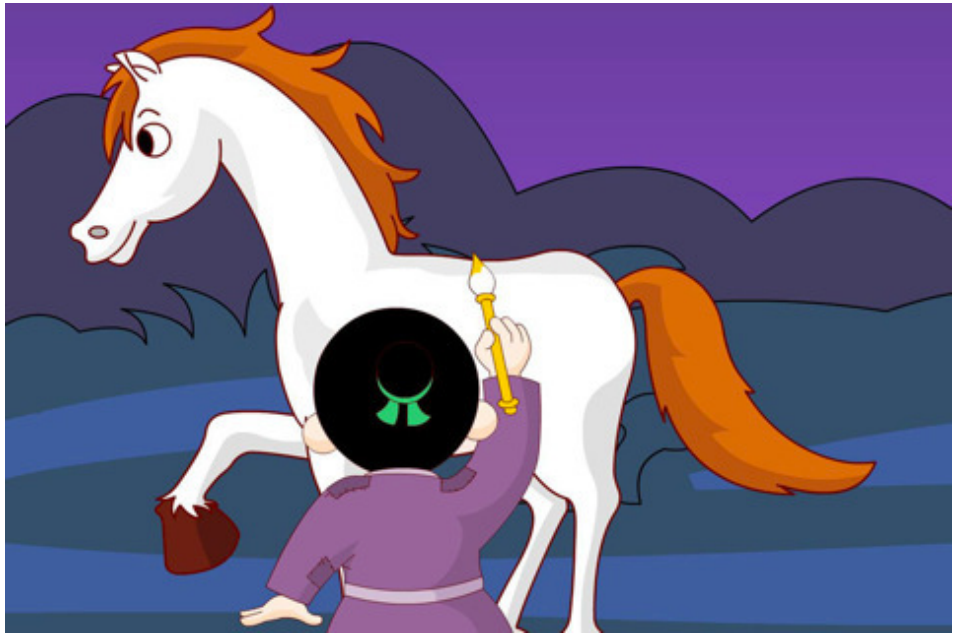


- Identified by the tuple ***(src IP, dest IP, src port, dest port, protocol id, type of service, interface)***
- Sampled every 5 minutes
- Contributed by the top-5 applications
- Account for about **80%** of the total traffic

Elephant flows

$$
\begin{array}{cccccc}
 & & \vdots & & & \\
in^{t+0} & out^{t+0} & ein_1^{t+0} & eout_1^{t+0} \ldots & ein_M^{t+0} & eout_M^{t+0} \\
in^{t+1} & out^{t+1} & ein_1^{t+1} & eout_1^{t+1} \ldots & ein_M^{t+1} & eout_M^{t+1} \\
in^{t+2} & out^{t+2} & ein_1^{t+2} & eout_1^{t+2} \ldots & ein_M^{t+2} & eout_M^{t+2} \\
 & & \vdots & & & \\
in^{t+s} & out^{t+s} & ein_1^{t+s} & eout_1^{t+s} \ldots & ein_M^{t+s} & eout_M^{t+s} \\
 & & \vdots & & & \\
in^{t+k} & out^{t+k} & ein_1^{t+k} & eout_1^{t+k} \ldots & ein_M^{t+k} & eout_M^{t+k}
\end{array}
$$

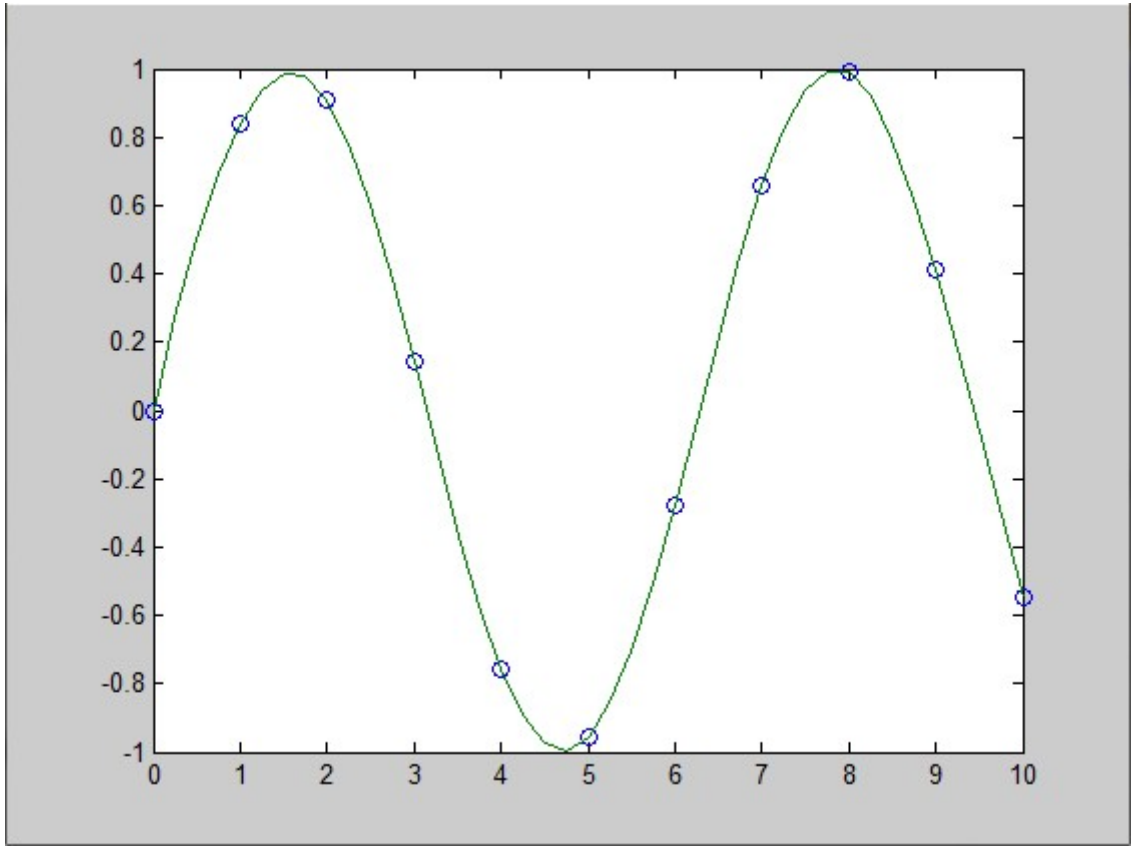
- Sampled every 5 minutes
  - Due to resource cost concerns
  - vs 30 seconds (total traffic)
- Interpolation
  - Construct missing values
  - Tried four methods



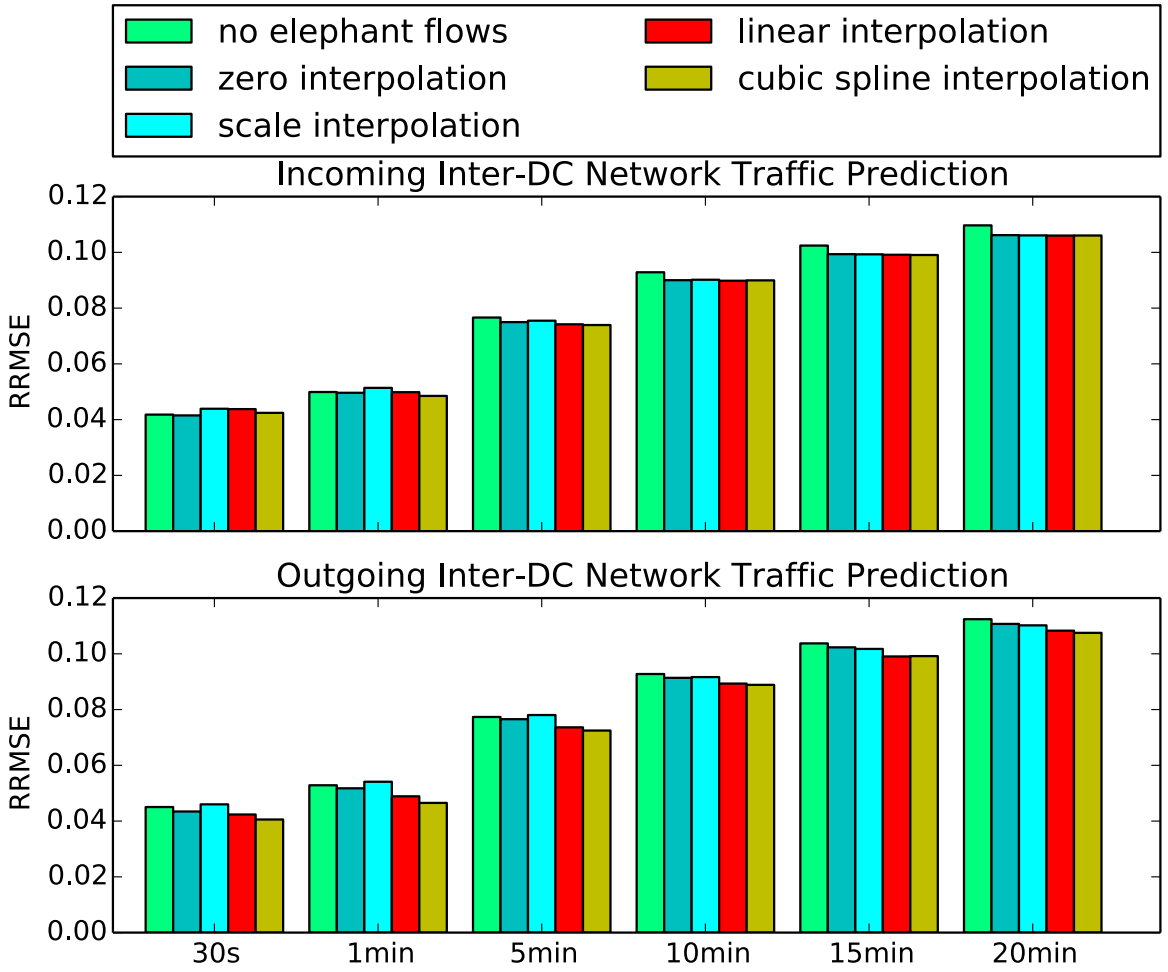
Elephant flows

Tried different Interpolation

- Zero interpolation
  - fill zeroes for unknown points
  - simplest
- Scale interpolation
  - fill numbers proportional to the total traffic
- Linear interpolation
  - a filled point is in a line segment linking the previous and the following points
- Spline interpolation
  - give a smooth curve linking points
  - third order polynomials as interpolation functions
  - error is small
  - most complex

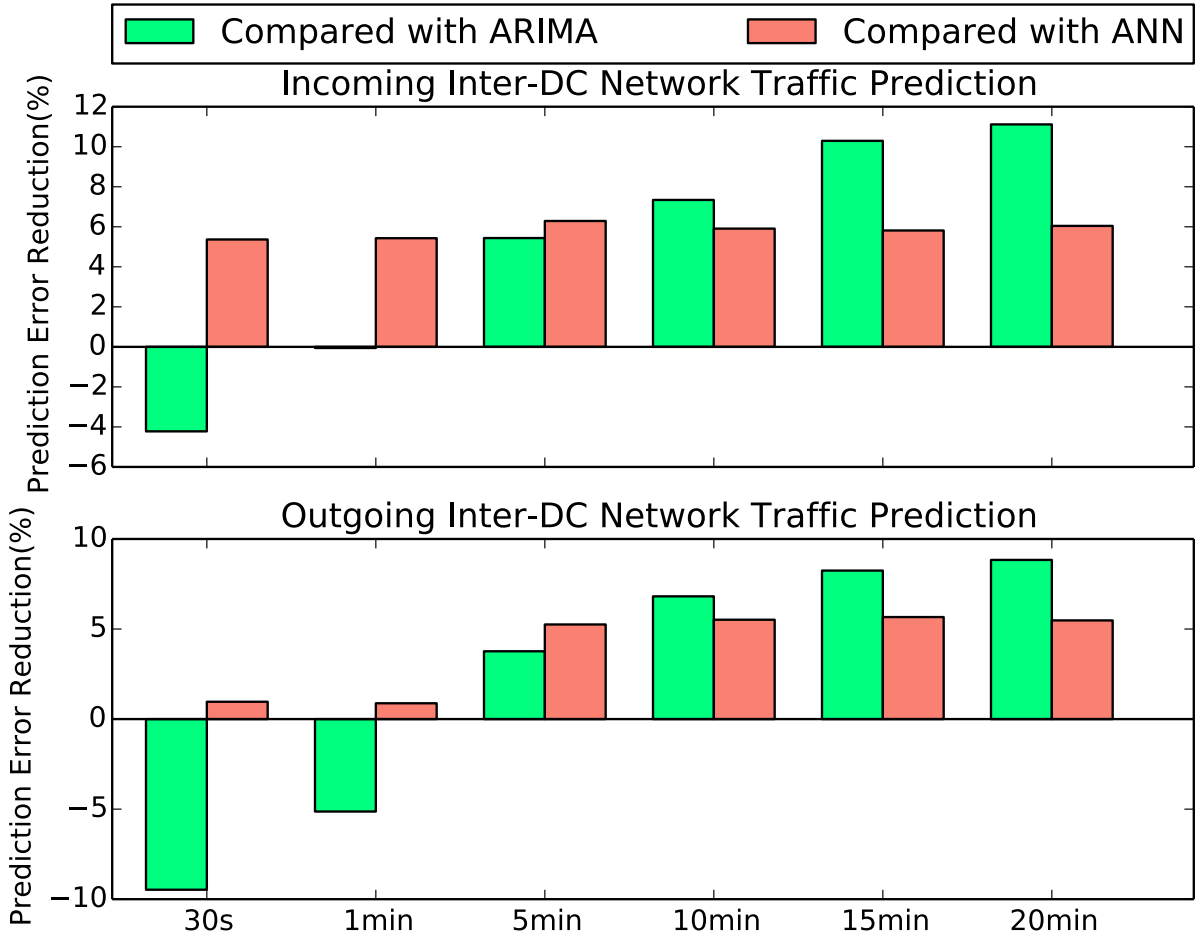


Elephant flows



- Elephant flow information reduces the prediction errors
  - for both incoming and outgoing traffic
  - especially for long-time prediction
- The more smooth the constructed curve is, the better the overall prediction accuracy is
  - cubic interpolation performs best
- Zero interpolation is chosen in production
  - different interpolation methods have similar effects
  - good balance between simplicity, practicability and performance

Compared with other models



- Compared with two well-known models: **ARIMA** and **ANN** without WT and elephant flows
- ARIMA performs the best only for very short prediction
- Our model performs the best for long-time-ahead prediction
  - It is difficult for linear models to capture long-term patterns of inter-DC traffic
  - Wavelet transform and elephant flow information are helpful for training with ANN

Conclusion and Future Work

- A new model for inter-DC network traffic prediction
  - treat elephant flow information explicitly

- Key idea: decompose the various components from the combined traffic pattern
  - the wavelet transform is an **internal decomposition**
  - separating out the elephant traffic can be treated as an **external decomposition**

- Practical considerations: reduce production cost
  - reduce the flow sampling overhead using interpolation methods
  - reduce the training overhead by 40% by combining incoming/outgoing traffic
  - Accurate prediction => reduce the peak bandwidth for about 9% for Baidu

- Future Work
  - Predicting longer period trends (weeks to months)
  - Models on multiple inter-DC link prediction

Yi Li
Tsinghua University
li-yi13@mails.tsinghua.edu.cn