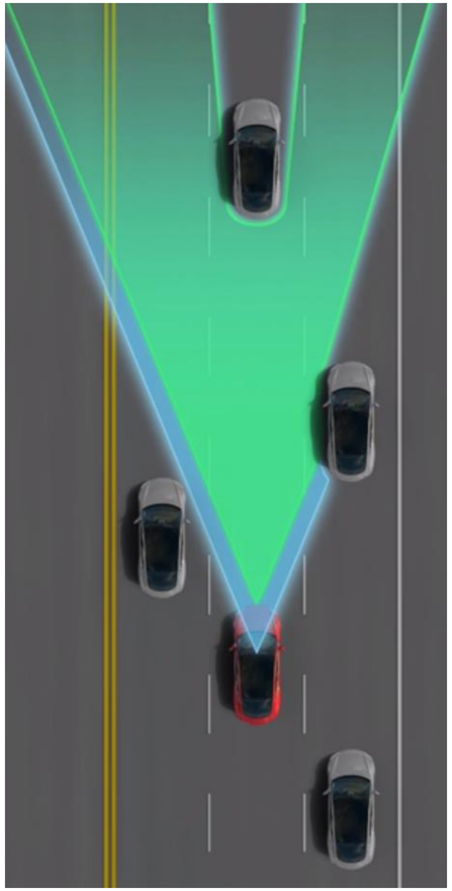
Query-based Workload Forecasting for Self-Driving Database Management Systems

Lin Ma, Dana Van Aken, Ahmed Hefny, Gustavo Mezerhane,
Andrew Pavlo, Geoffrey J. Gordon

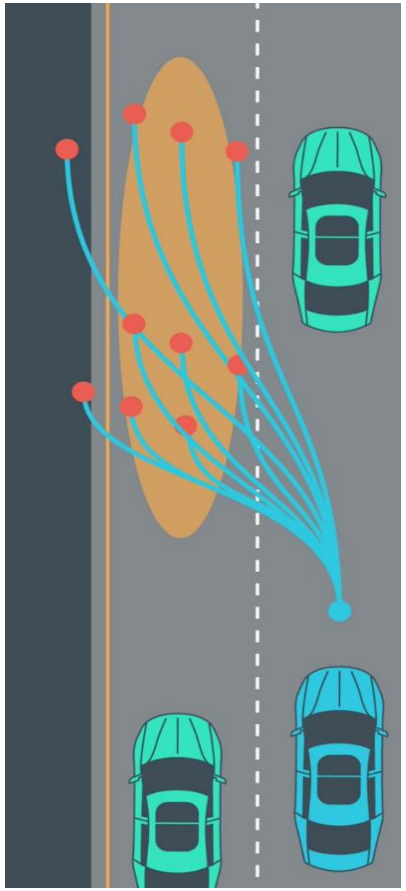
Self-Driving Cars

Perception



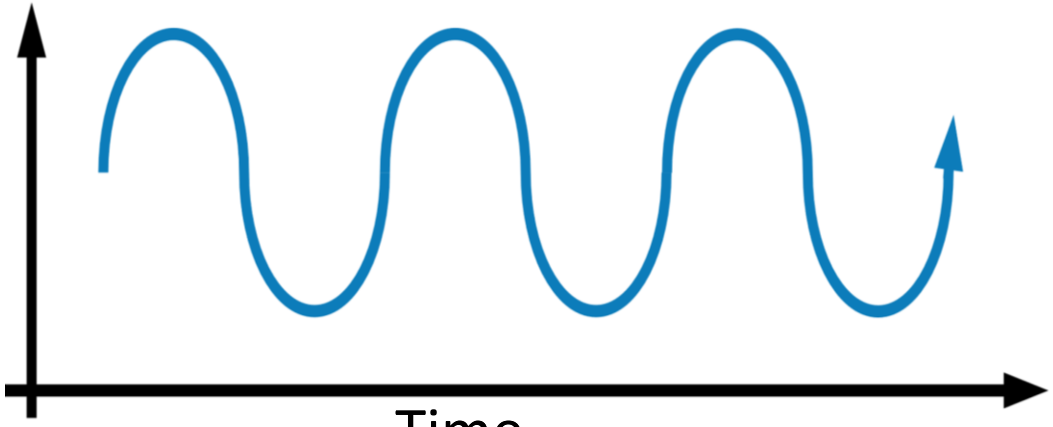
Actions

Planning

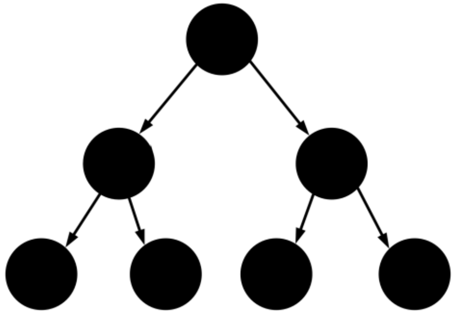


First Step Towards Self-Driving Databases

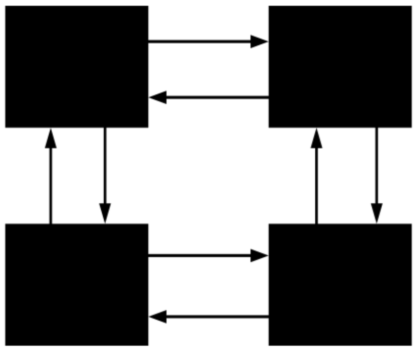
Workload Forecasting



Time



Indexing

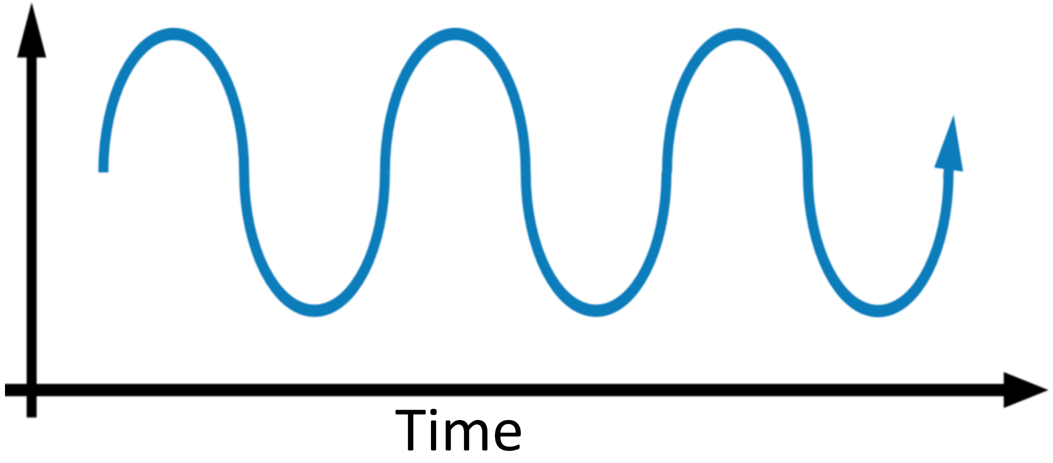


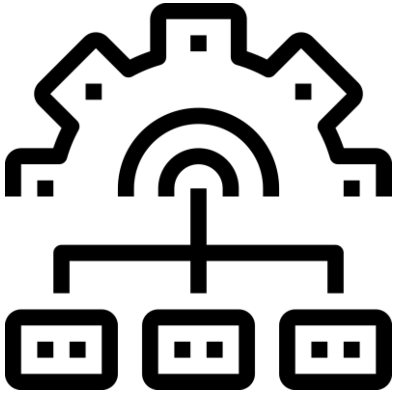
Partitioning

Scaling

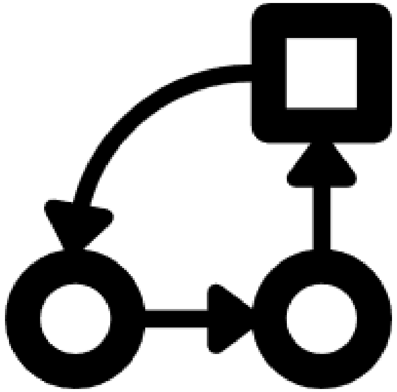
First Step Towards Self-Driving Databases

Workload Forecasting





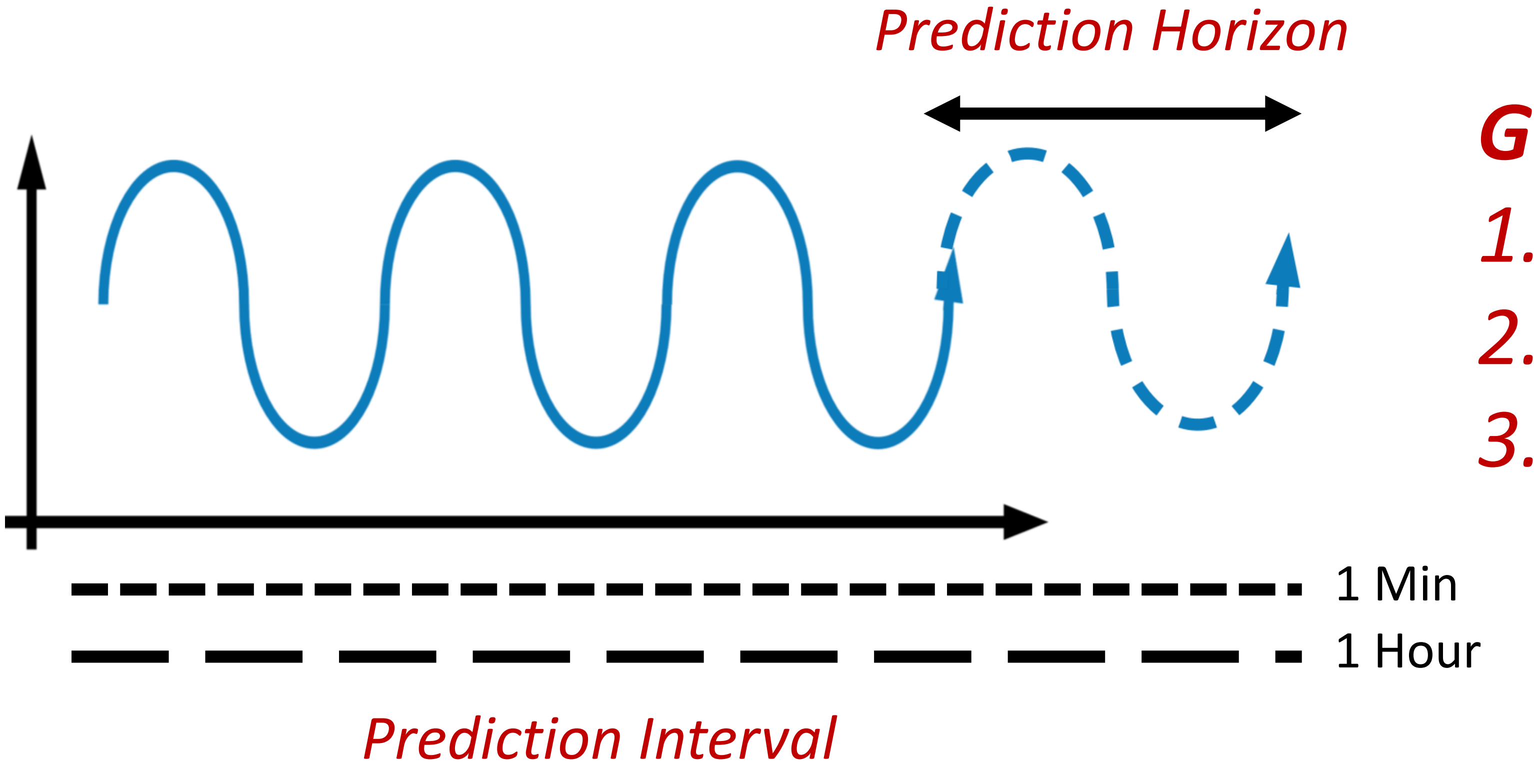
Planning



Observation

Workload Forecasting

- When, how many, and what queries will arrive



Prediction Horizon

1 Min

1 Hour

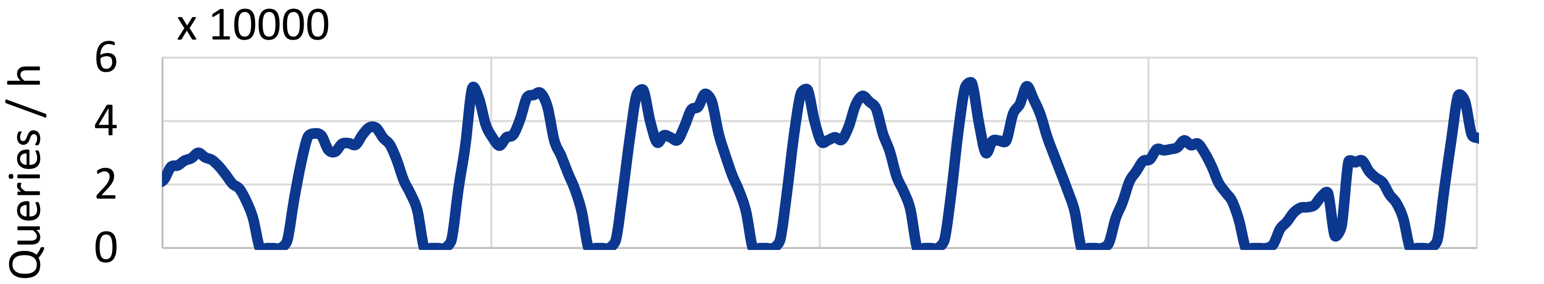
Prediction Interval

Goals:

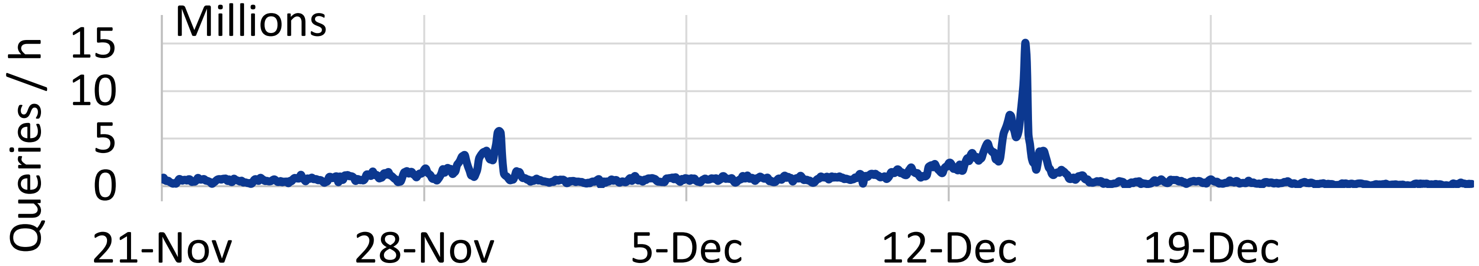
1. *Good Accuracy*
2. *Major Patterns*
3. *Cost vs. Accuracy*

Challenges

- Support for dynamic workloads
- Support large query volumes
  - *Millions / Day*
- Support different arrival rate patterns

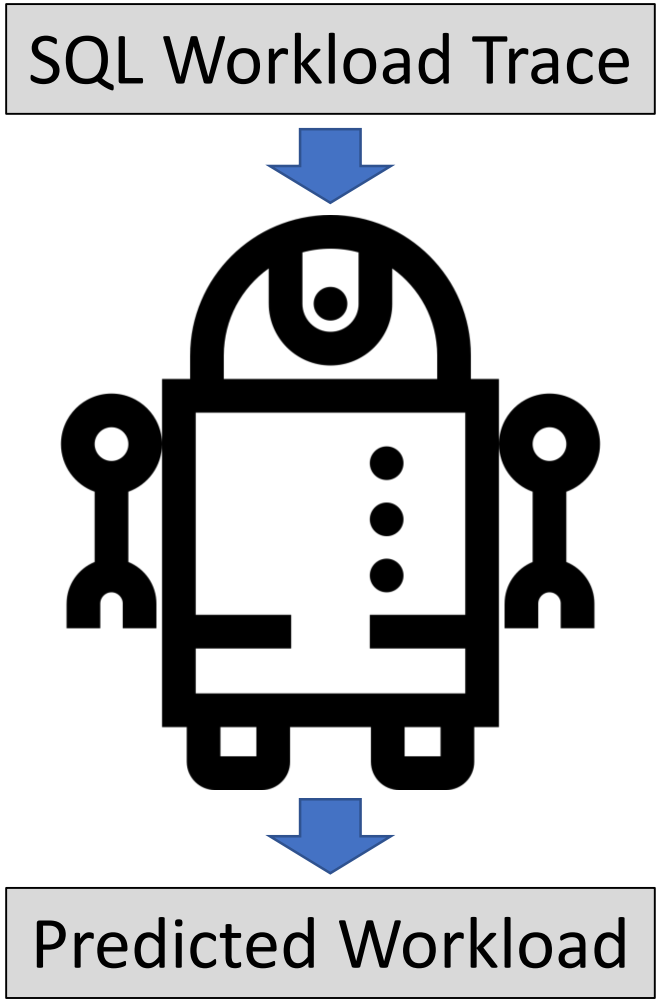


Cyclic (Diurnal)



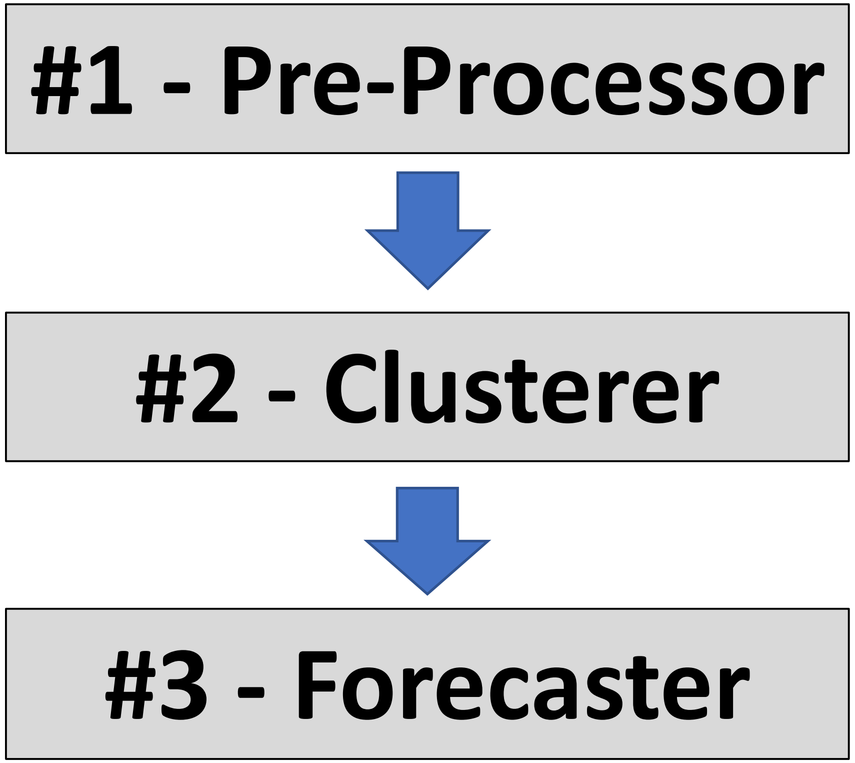
Growth and Spike

QueryBot 5000



SQL Workload Trace

Predicted Workload



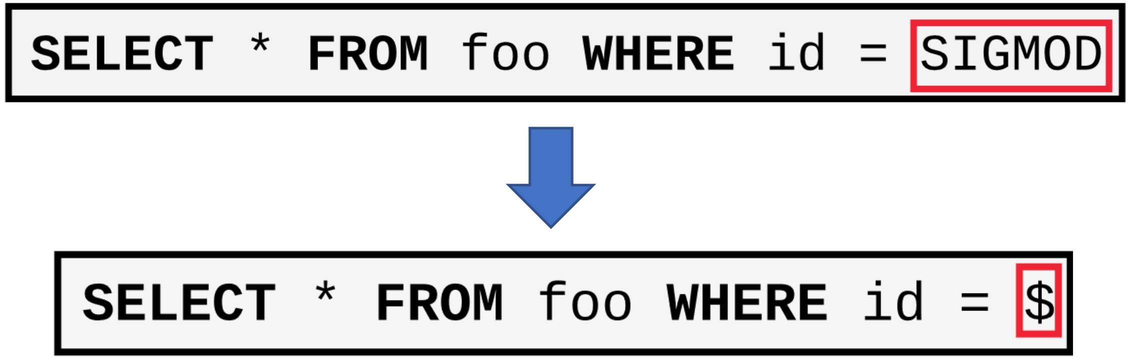
#1 - Pre-Processor

#2 - Clusterer

#3 - Forecaster

Step #1 - Pre-Processor

- Templatization



```
SELECT * FROM foo WHERE id = SIGMOD
```

```
SELECT * FROM foo WHERE id = $
```

- Semantics equivalence check

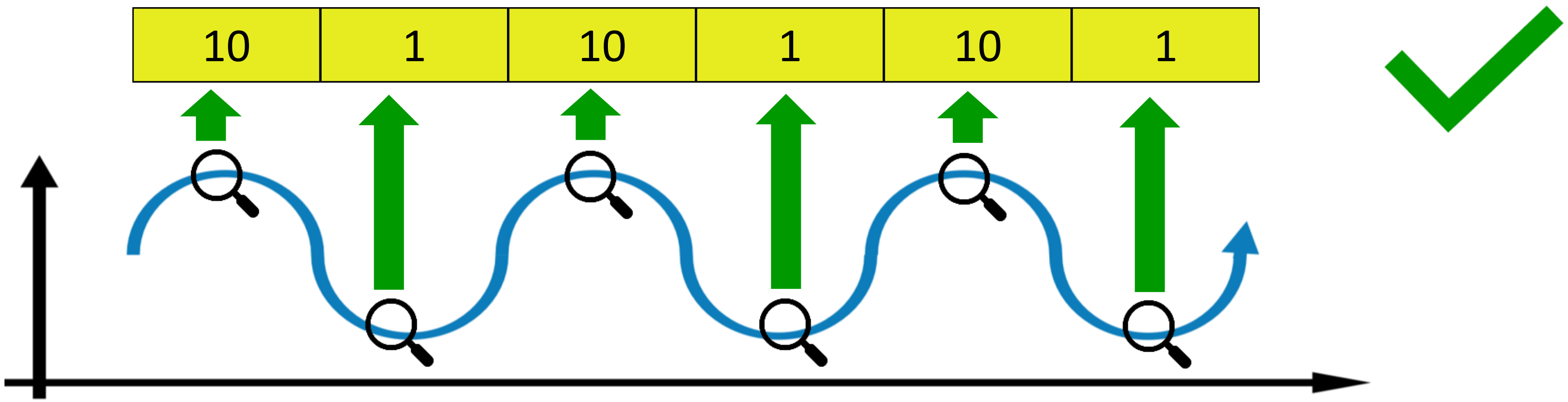
Millions → Thousands 😏

Step #2 - Clusterer

- Possible Similarity Features
  - *Physical Feature*
  - *Logical Feature*
  - *Arrival Rate Feature*

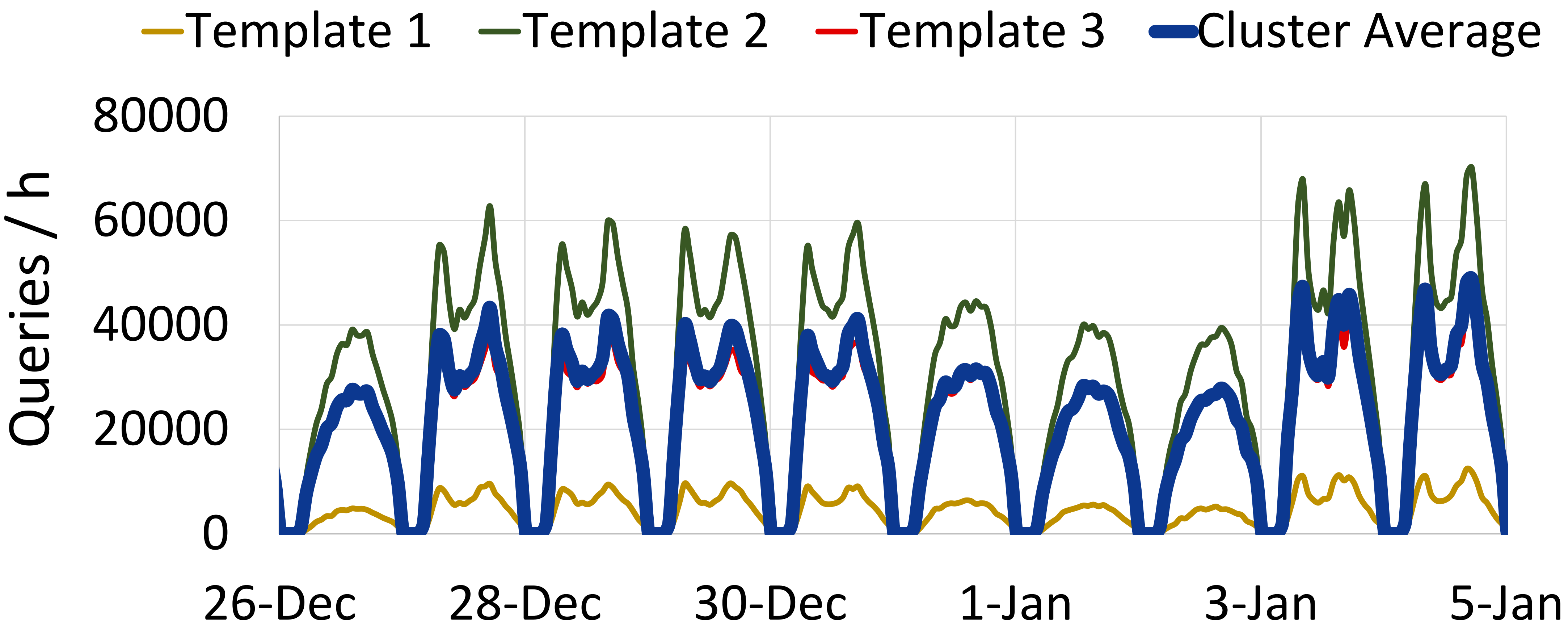
# Tuples Read	~~# Tuples Write~~	Latency	…

Query Type	~~Columns Referenced~~	# JOINs	…



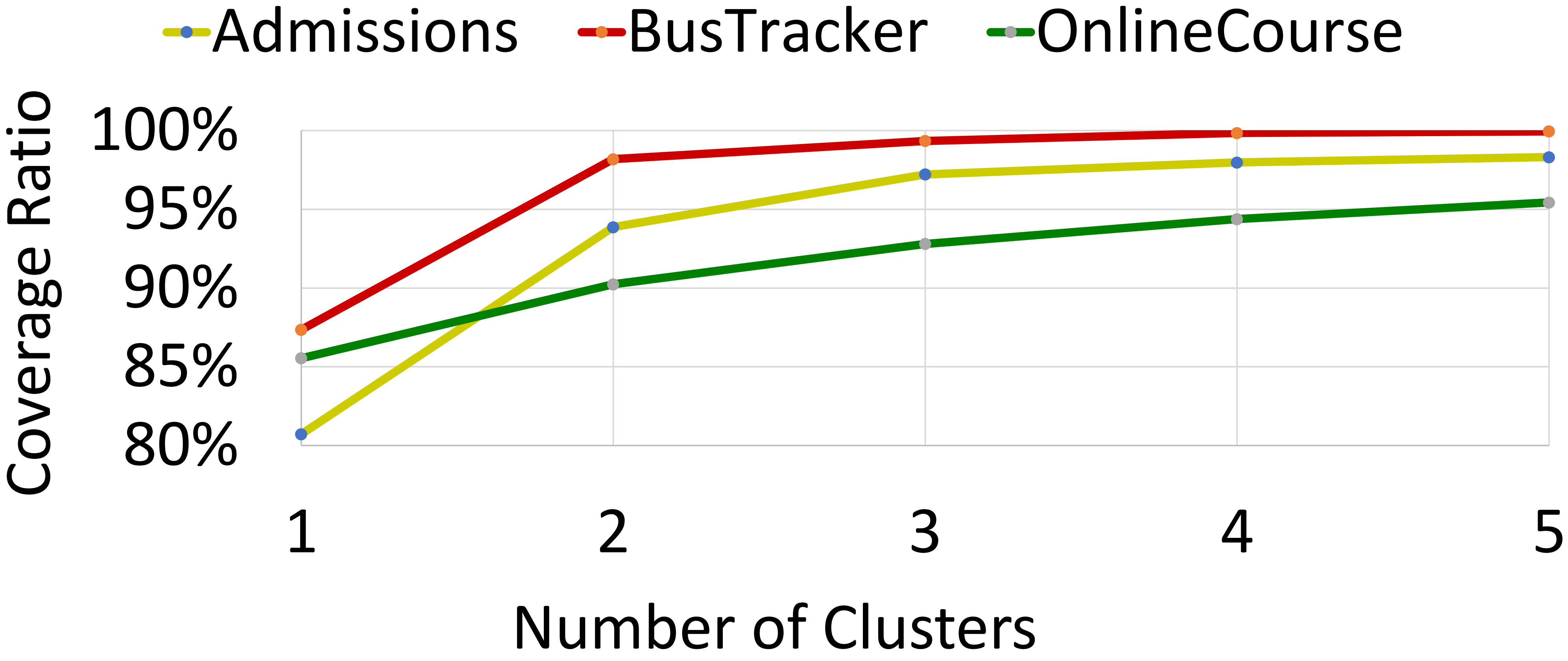
10	1	10	1	10	1

Arrival Rate History



Bus Tracking App

Coverage of the Largest Clusters



A few large clusters exhibit major patterns

Step #3 - Forecaster

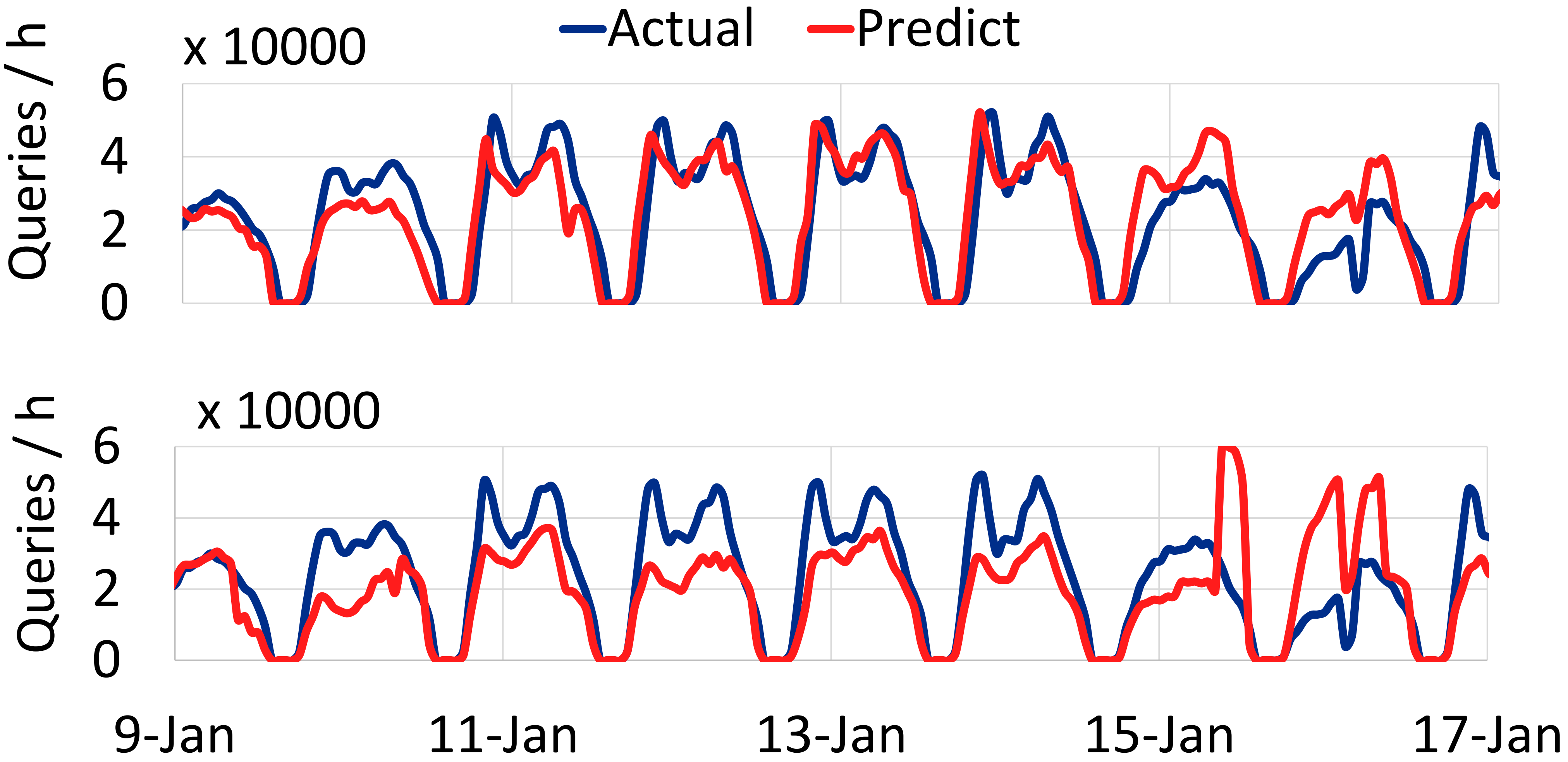
- Different models have different properties
  - *Linear Regression (LR), ARMA, Kernel Regression (KR), Recurrent Neural Network (RNN), FNN, PSRNN*
  - *Properties: Linear, Memory, Kernel*
- Ensemble: combine different models

LR+RNN has the best average accuracy

Prediction Results

1 Hour Horizon:

1 Week Horizon:

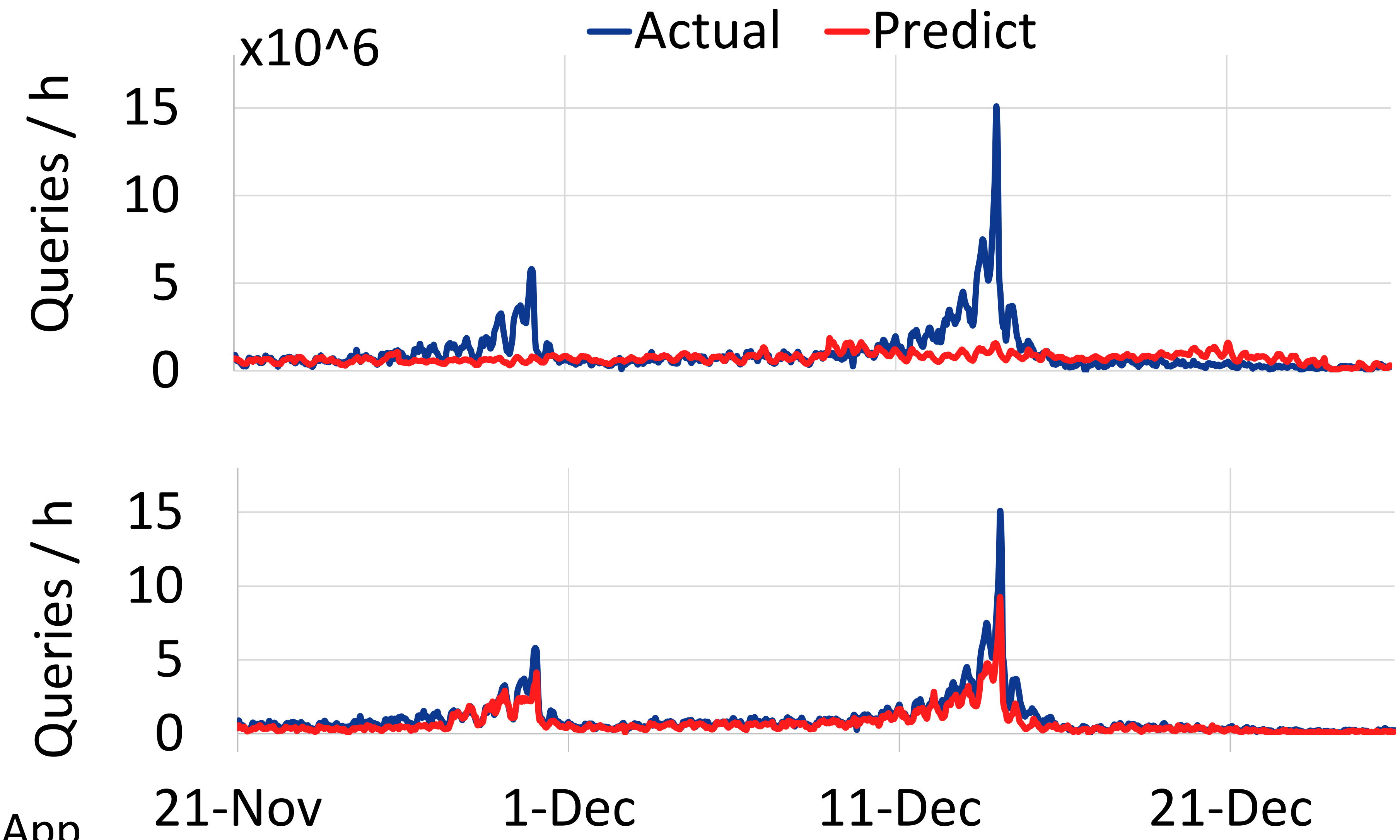


Bus Tracking App

Prediction Results for Spikes

ENSEMBLE:
(LR+RNN)

KR:

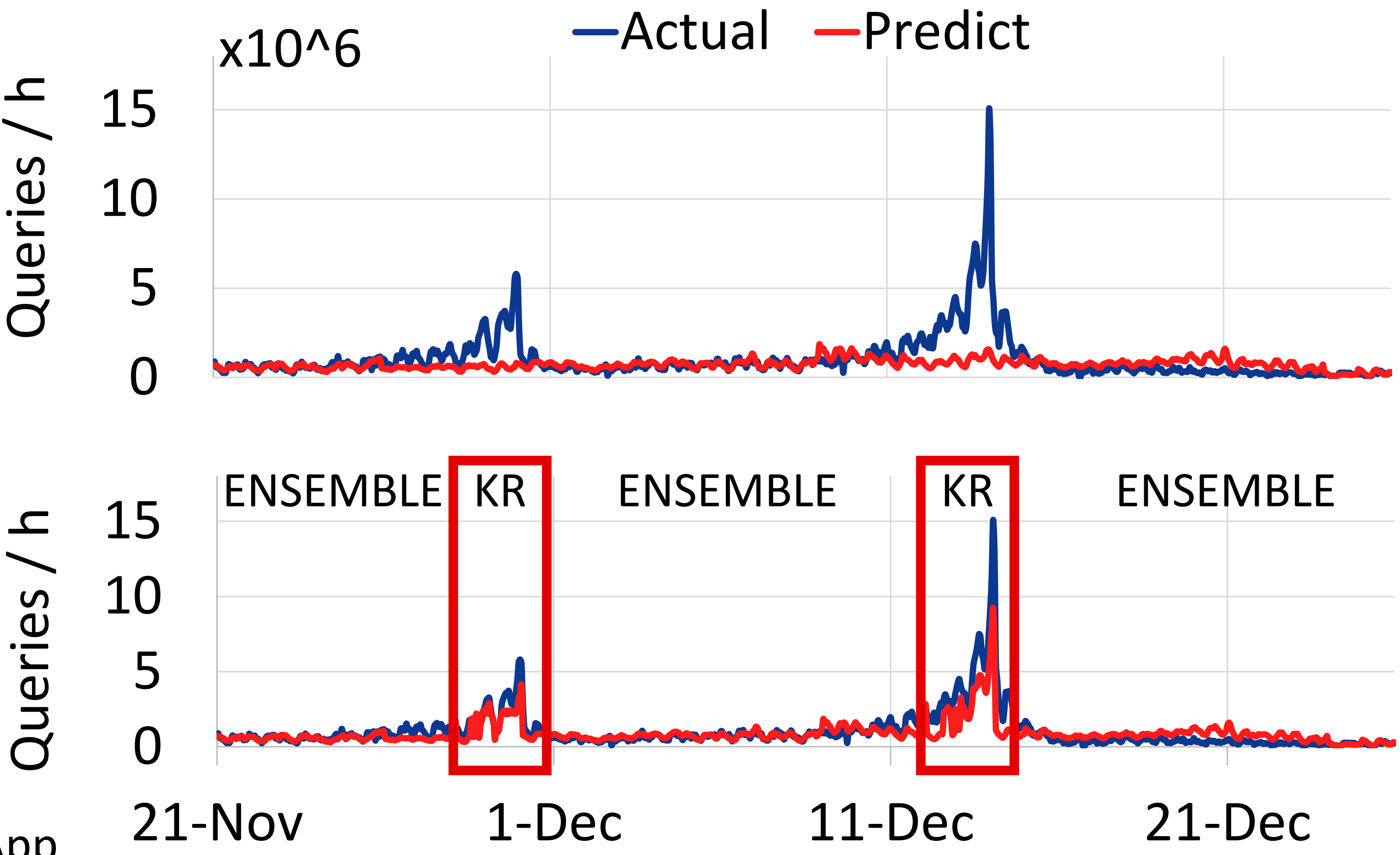


Admissions App
1 Week Horizon

HYBRID Model

ENSEMBLE:
(LR+RNN)

HYBRID:
(LR+RNN+KR)

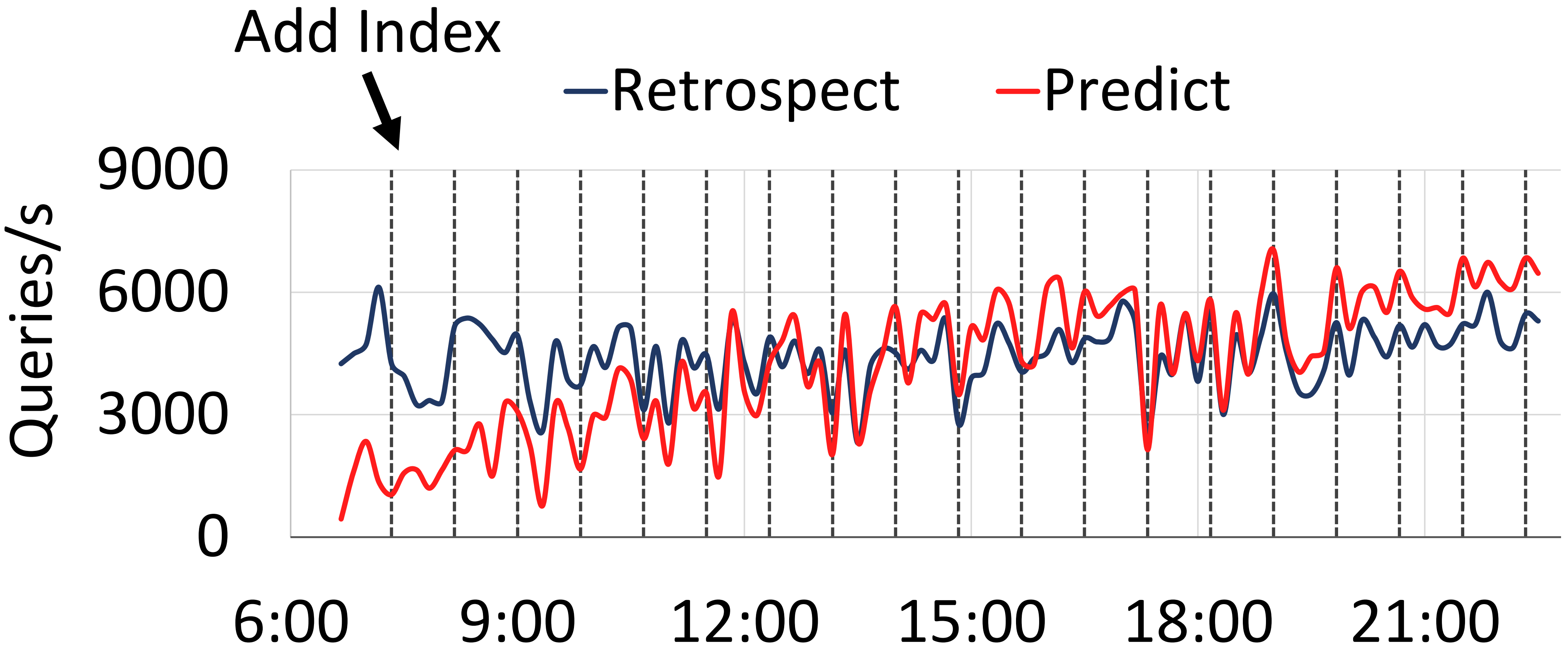


Admissions App
1 Week Horizon

Example: Automatic Index Building

- Integrate QB5000 with MySQL
- Start with only primary indexes
- Same index suggestion algorithm to build 20 indexes -
  - *RETROSPECT: Build all indexes at once with sample history*
  - *PREDICT: Build indexes one at a time using the forecasting*

Example: Automatic Index Building



Admissions App

Takeaways

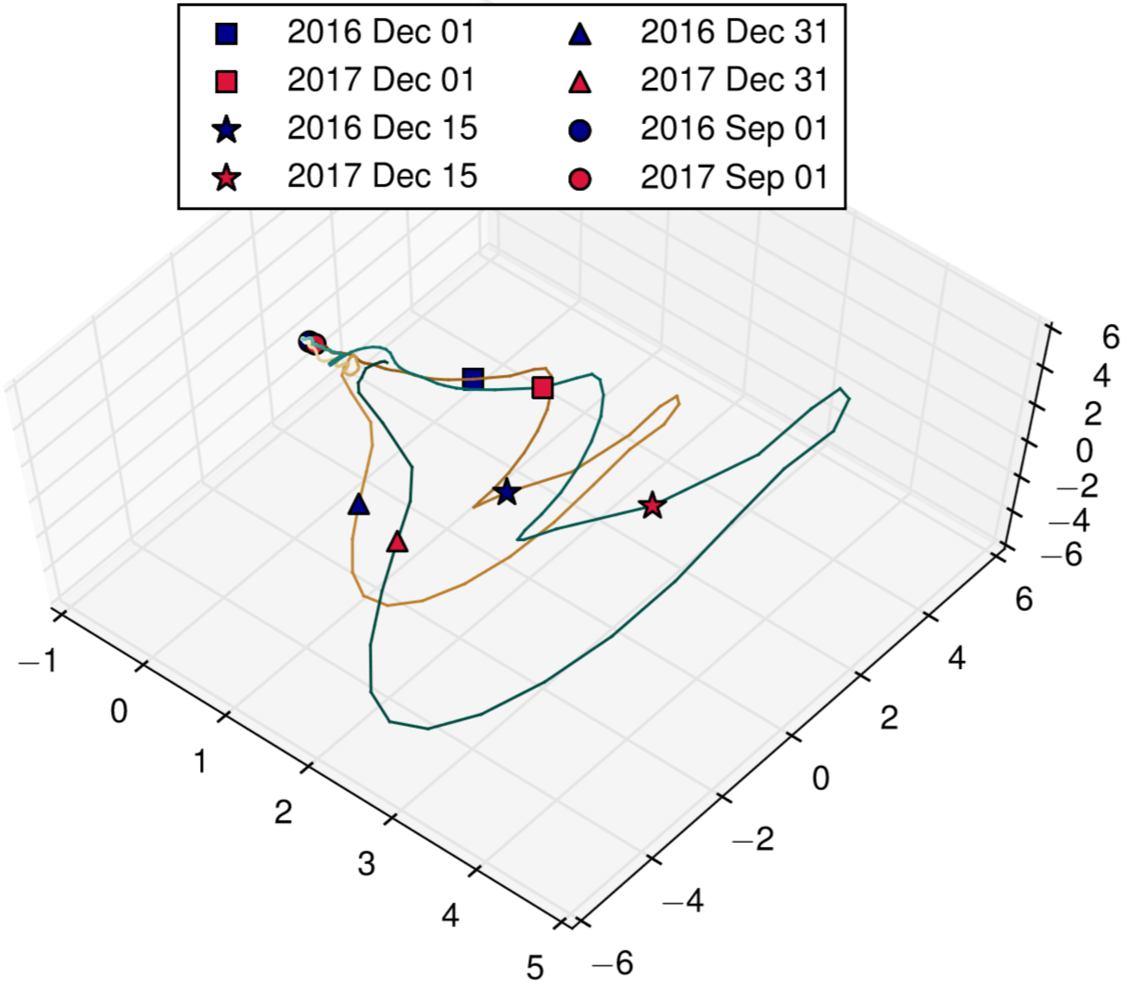
- Workload forecasting on combinations of horizons/intervals
- Reduce the forecasting cost with minimal lost of accuracy
  - *Templatization*
  - *Clustering on arrival rate feature*
- Hybrid forecasting method
  - *Capture major database workload patterns*
  - *Maintaining good average accuracy*

END

lin.ma@cs.cmu.edu

https://github.com/malin1993ml/QueryBot5000

Input Space For Kernel Regression



Related Work

- Resource Demand Prediction
- Performance Modeling and Diagnosis
- Next SQL/Transaction Prediction
- Workload Shift Detection
- Workload Compression
- Run-time Metrics Prediction (e.g. latency)