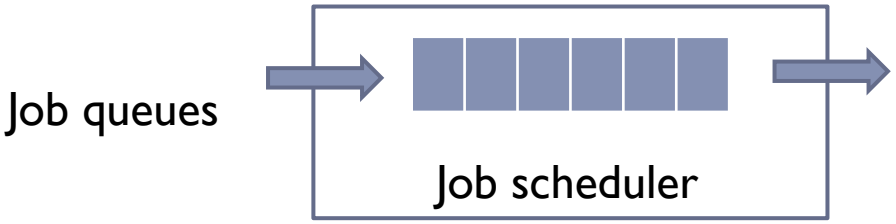
Predicting Job Completion Times Using System Logs in Supercomputing Clusters

Xin Chen, Charng-Da Lu and Karthik Pattabiraman
University of British Columbia (UBC)

A Challenge in Supercomputing Systems

- Failures in supercomputers
- Challenges for applications
  - Complete tasks
  - Achieve high throughput
- Possible solutions
  - Workload management
  - Checkpointing and recovery

http://ccr.buffalo.edu/support/research_facilities.html



System Log Analysis

- Indicators of a system's health status
  - Most produced during normal operations
  - Isolate log messages that are indicative of job terminations
- Untagged system logs

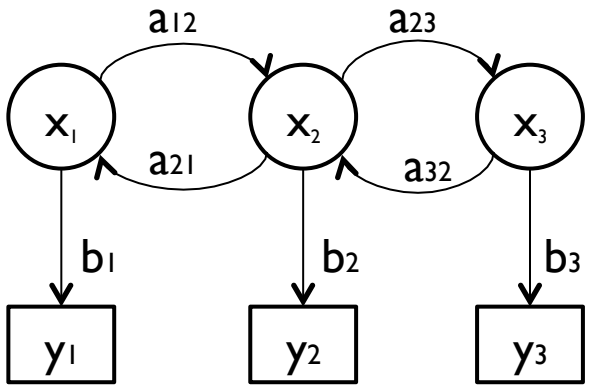
kernel: blcr: Checkpoint/Restart module removed
kernel: imklog, log source = /proc/kmsg started

- Tagged system logs (e.g., Blue Gene/L)

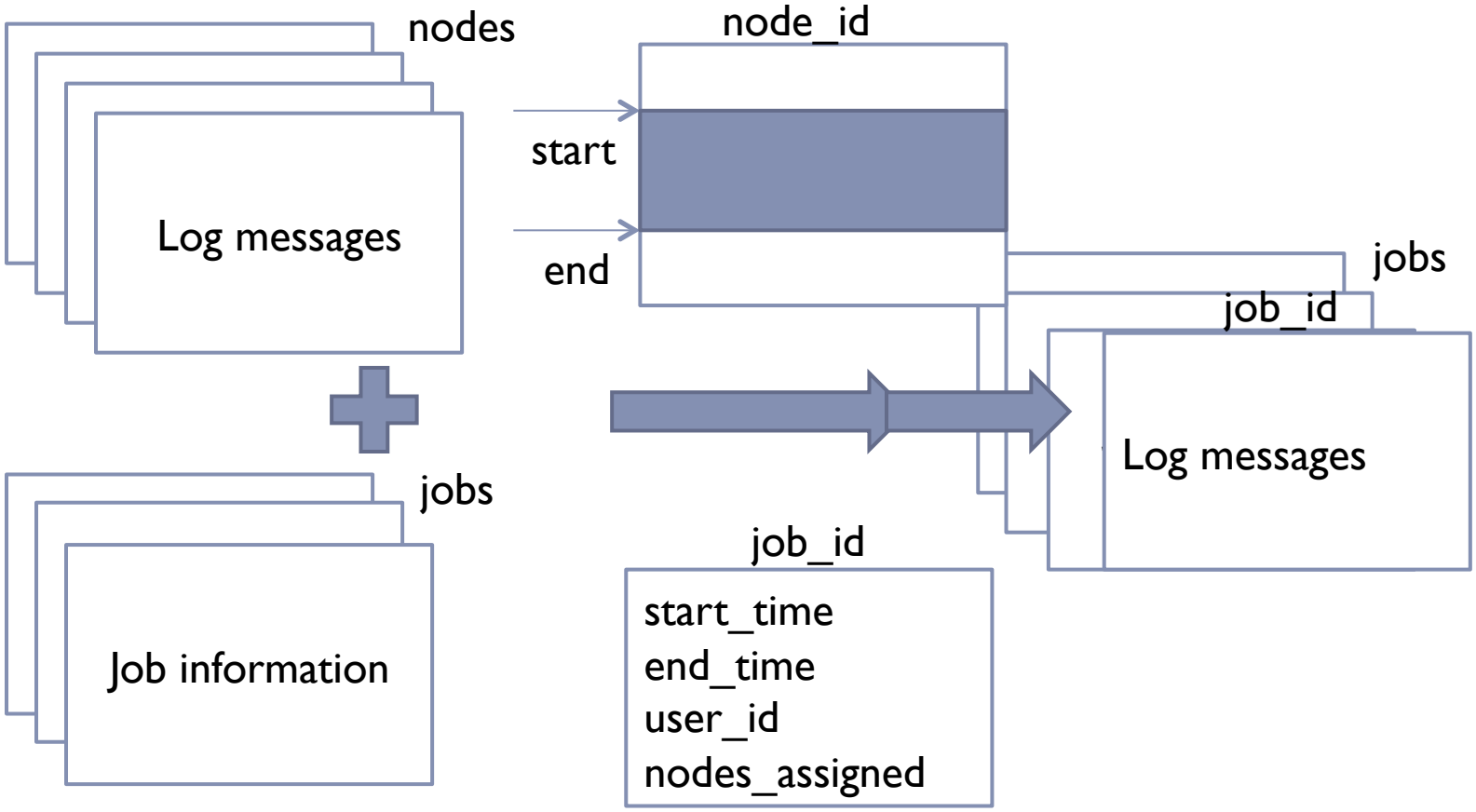
RAS KERNEL FATAL data TLB error interrupt
RAS KERNEL INFO instruction cache parity error corrected

Contribution

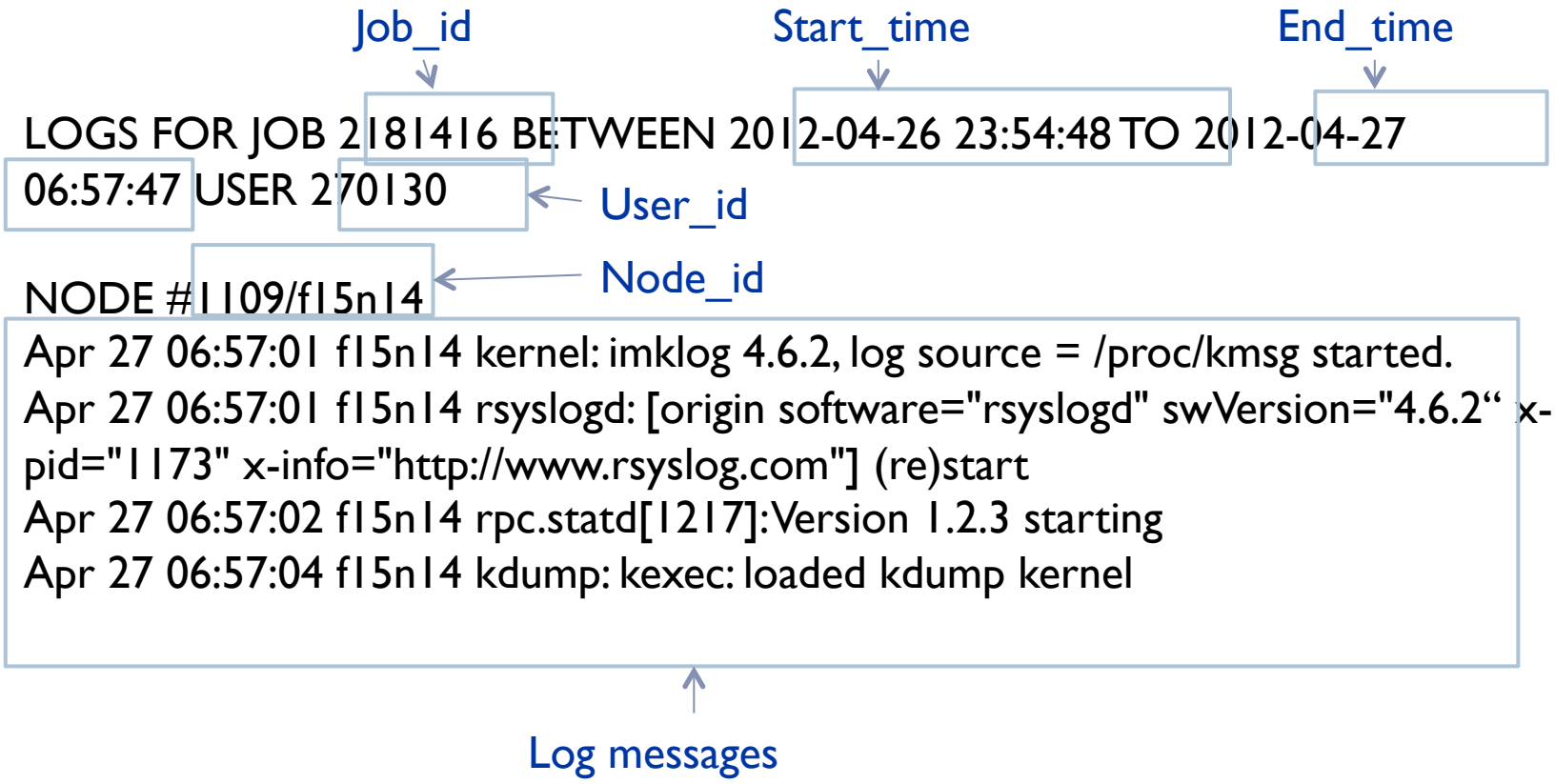
- Predict the job completion time
  - No annotations or tags about failures
  - A low rate of false positives
  - Use Hidden Markov Models (HMMs) to learn job running status



Job Sorted Log Messages

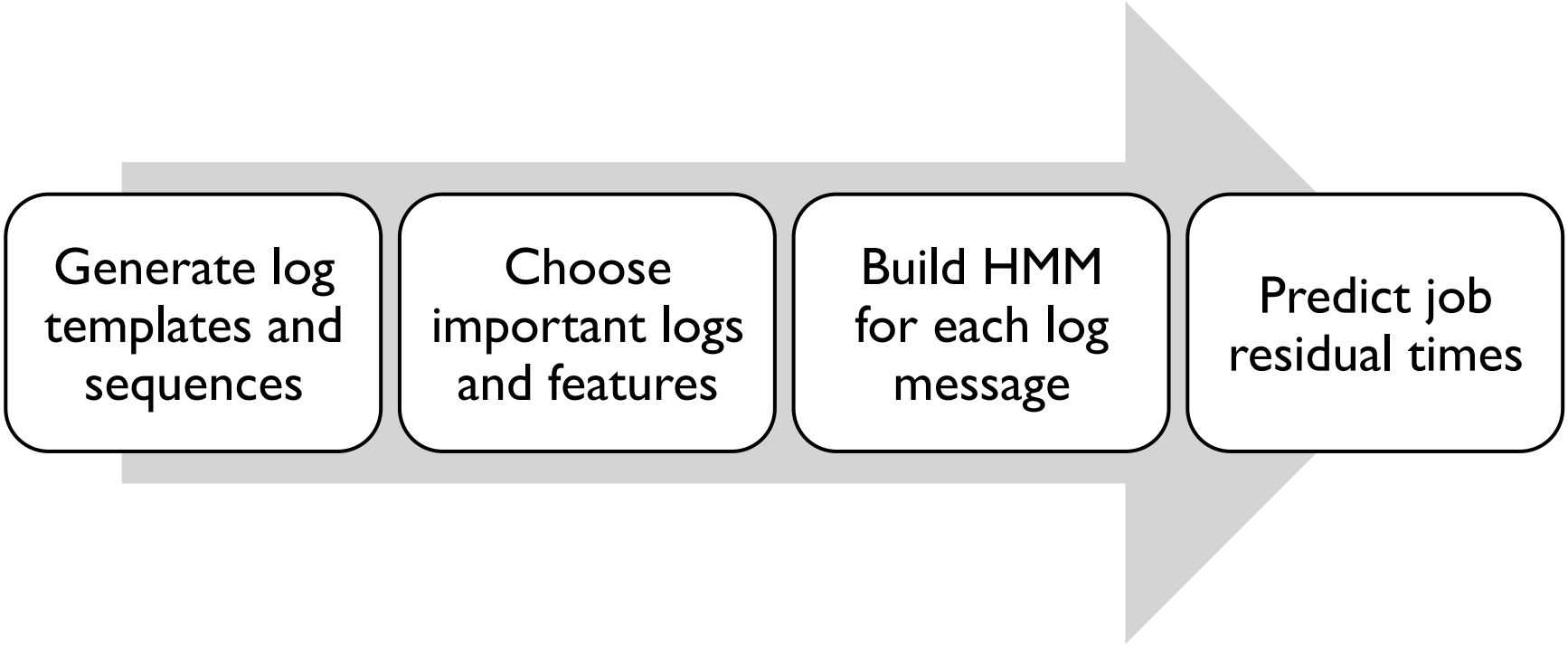


An Example of Part of a Job's Log Messages



Approach

- The prediction workflow



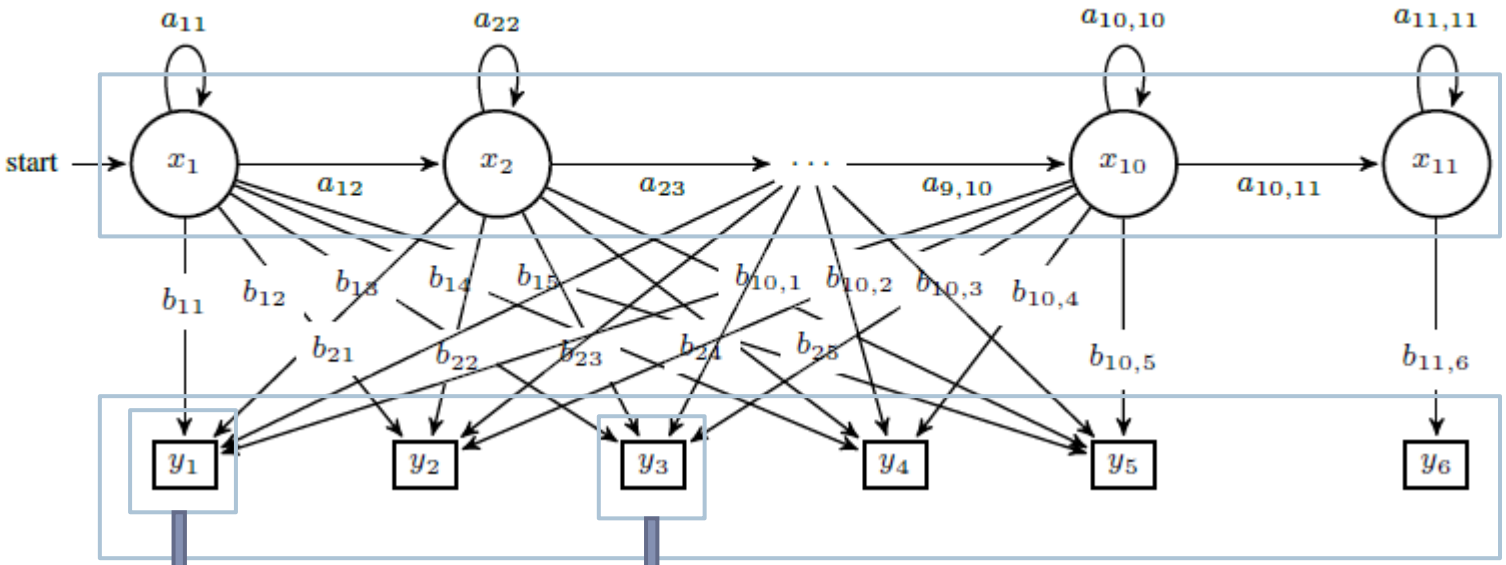
Important Log Templates

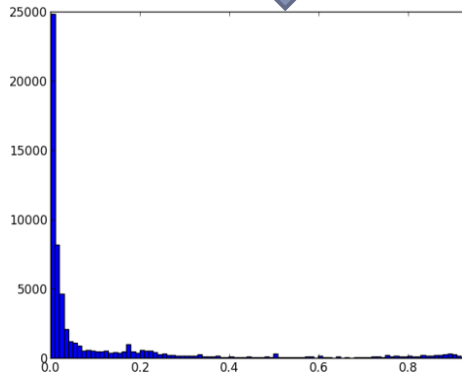
- A two-step process
  - Identify messages that are more likely to occur towards the end of a job

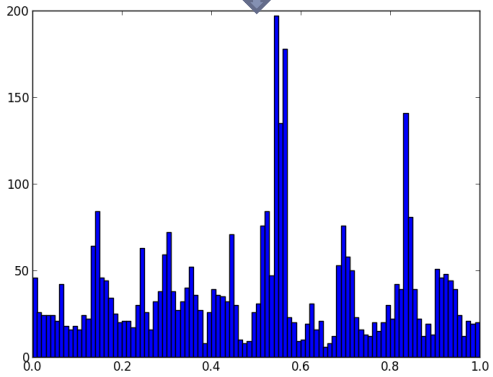
Example Log Template	Frequency
puppet-agent: content change	0.908669
pbs mom: req cpyfile Unable to copy file	0.939497
abrtd: no proper key	0.846774

  - Identify message pairs consisting of messages identified in the first step

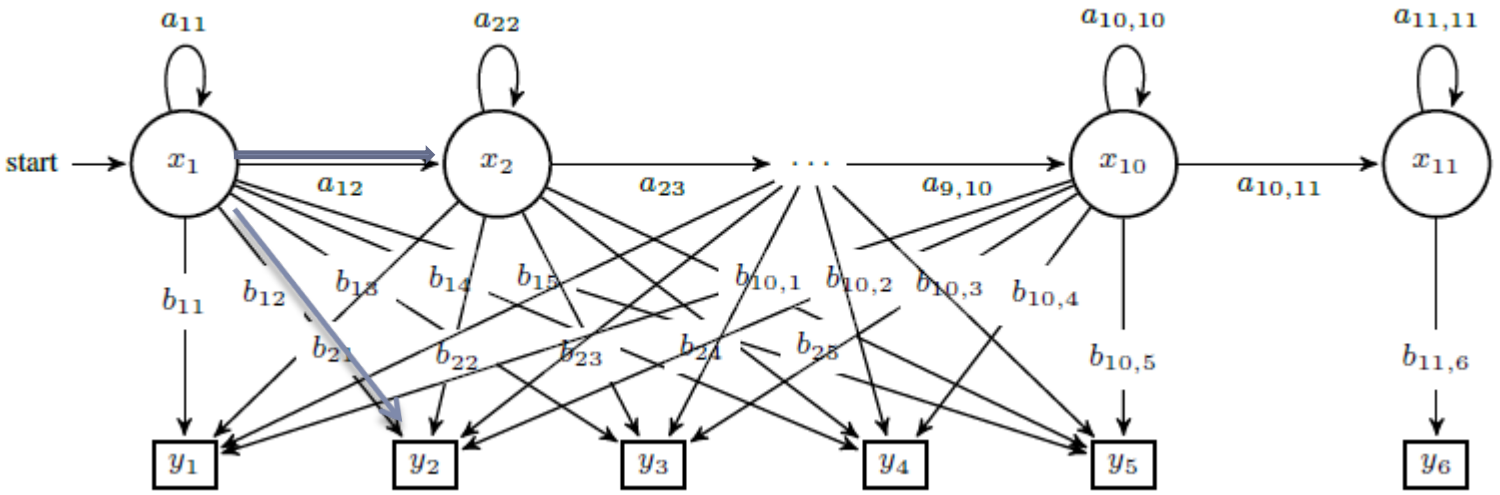
Build Hidden Markov Models







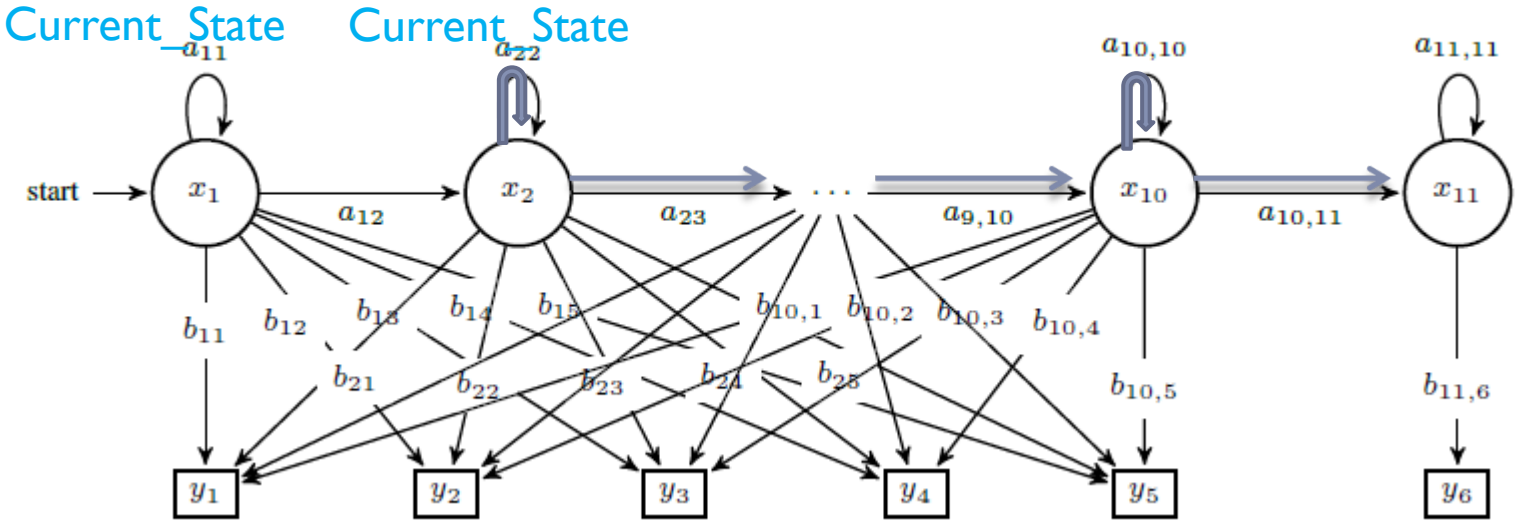
Build Hidden Markov Models



After a time step (10 seconds):

- Transitions between states
- Emissions from states to outputs

Residual Time Prediction

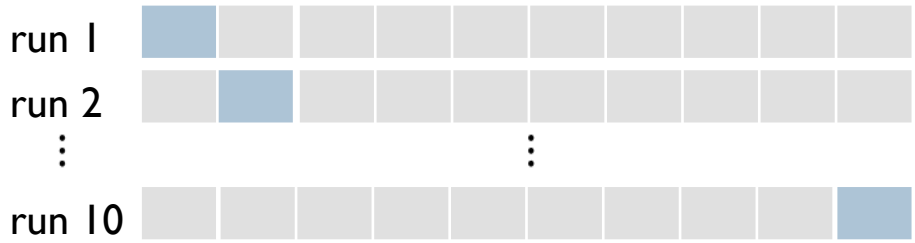


- Start from the first state, and compute HMM parameters
- Calculate the new model when an important log appears
- Estimate the most possible states in the period (Viterbi algorithm)
- Calculate the predicted residual time at the new state

Evaluation

- Dataset
  - The "Edge" cluster in Center for Computational Research at the State University of New York (SUNY) Buffalo
  - Logs in April 2012
  - 951 compute nodes and 120,639 jobs

- Cross validation
  - 10 folds: random and equal size



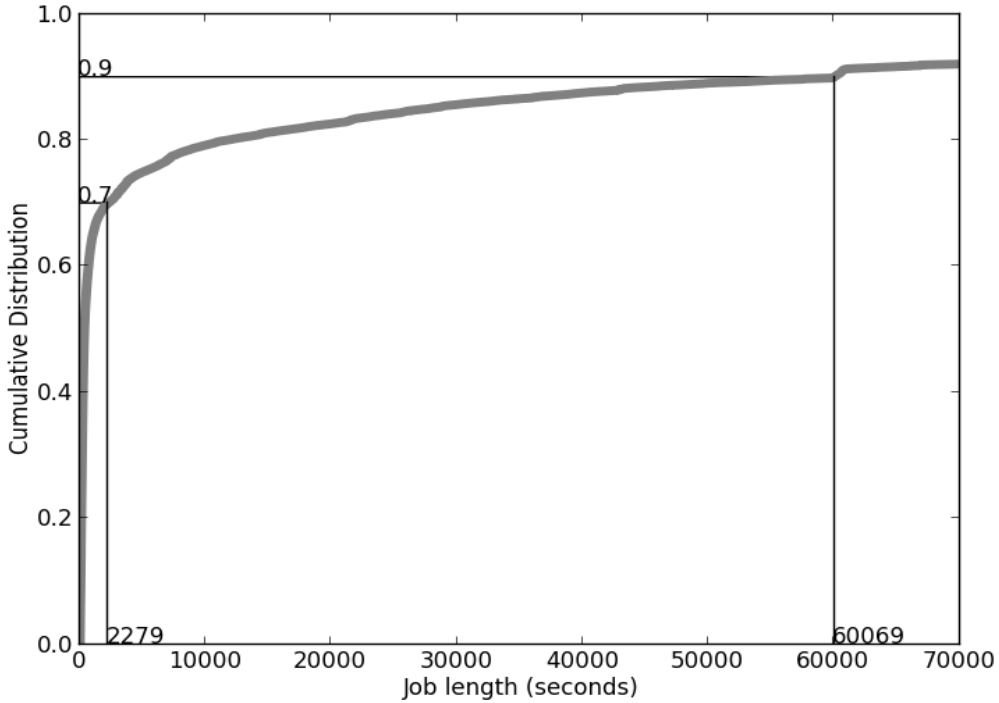
Results

- Errors of predicted and actual completion times
- The accuracy of predicting job termination within a short period

Prediction Time Errors

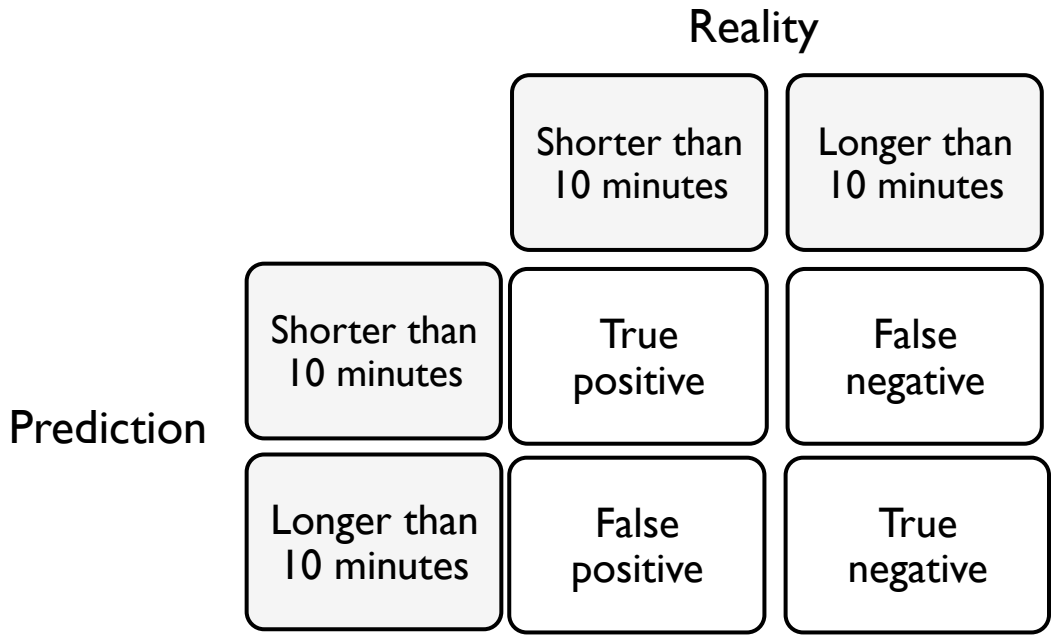
- Error between predicted and actual residual times
  - 91.28% prediction errors – less than 5000 seconds

    74.43% prediction errors – less than 200 seconds
- Job statistics



Predicting Job Termination Within a Short Period

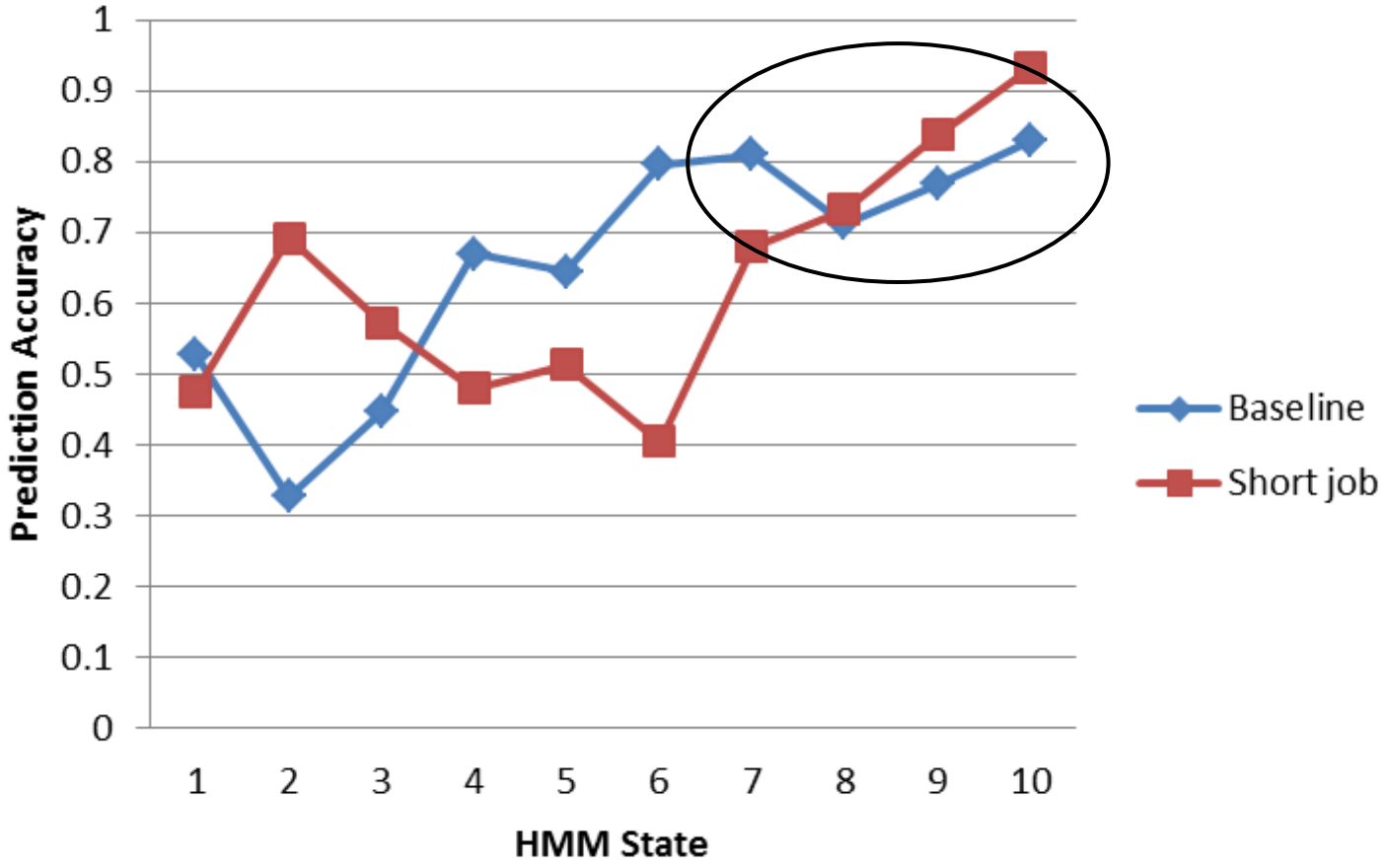
- Standard choice: 10 minutes



$$prediction\ accuracy = \frac{true\ positive + true\ negative}{true\ positive + false\ negative + false\ positive + true\ negative}$$

Predicting Job Termination Within a Short Period

- "Baseline model": the entire training datasets
- "Short job model": trained by jobs less than one hour



Conclusion

- Predicting job completion times in supercomputing clusters
  - Hidden Markov Models
  - Frequency-based log messages
- Predict 75% of jobs within 200 seconds of error
- Predicting job termination within a short period using short jobs train the HMM
  - A highest accuracy of 93% in the final states

- Future directions
  - Online prediction system
  - Mine and identify log subsequences and patterns