Intent Classification

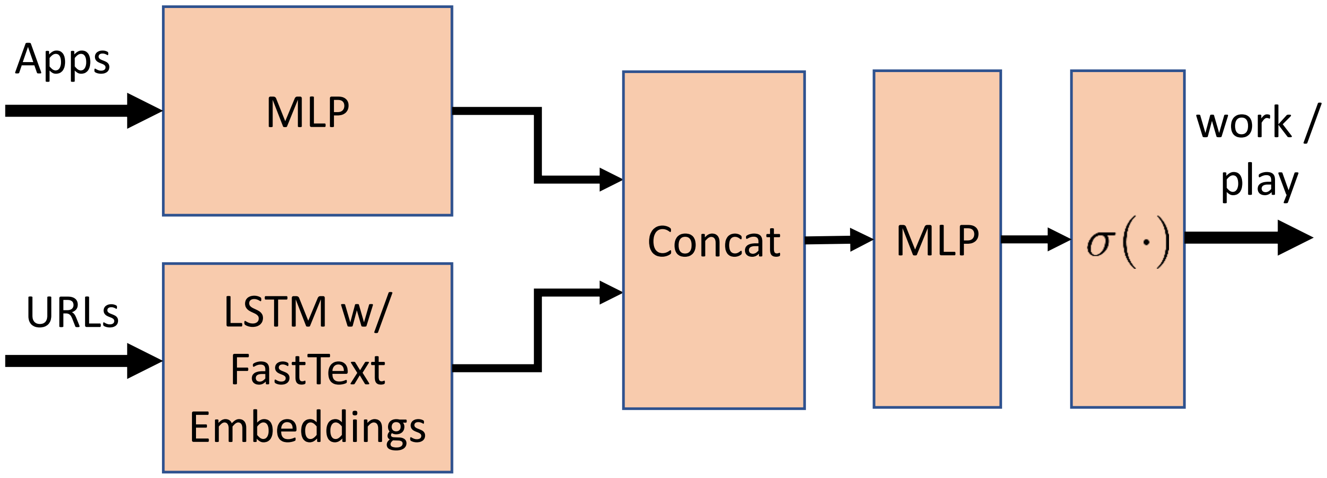
Intent Classification

Goals:

- Work / Play Classification based on URL and App data
- Smarter OS as a result of information sharing between the Apps: Next Session Type and Start Time prediction
- Anomaly Detection

Proposed Model:

- Multi-Modal Neural Network



- Advantages:
  - Scalable to multiple input branches and multiple modalities (e.g. images, text)
  - Input branches are designed independently but trained jointly

Data Exploration

URL data

- 1.5M rows, 15 categories
- Example (informative):

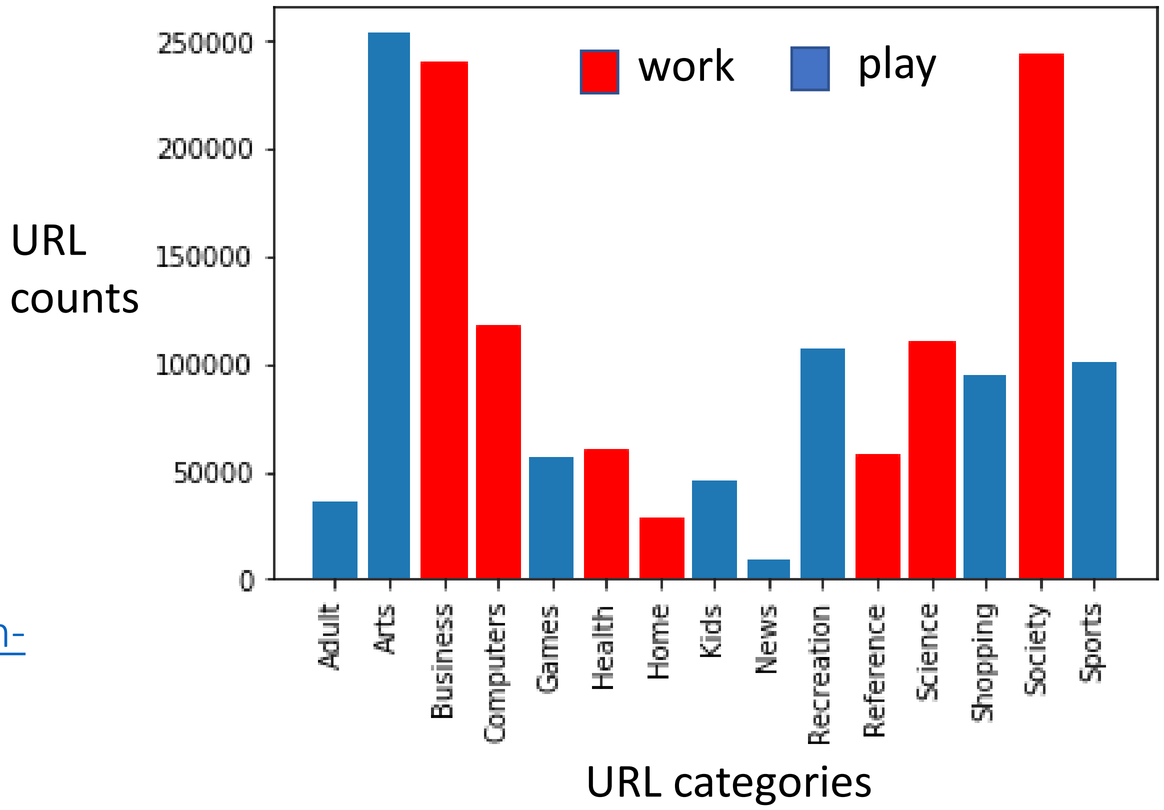
  original link:
  https://www.theverge.com/2019/7/1/20676939/nasa-orion-crew-capsule-launch-abort-system-test-emergency

  tokenized link:
  ['theverge', 'nasa', 'orion', 'crew', 'capsule', 'launch', 'abort', 'system', 'test', 'emergency']

- Example (uninformative):

  original link:
  https://www.google.com/search?q=test&oq=test&aqs=chrome..69i57j0l5.947j0j8&sourceid=chrome&ie=UTF-8

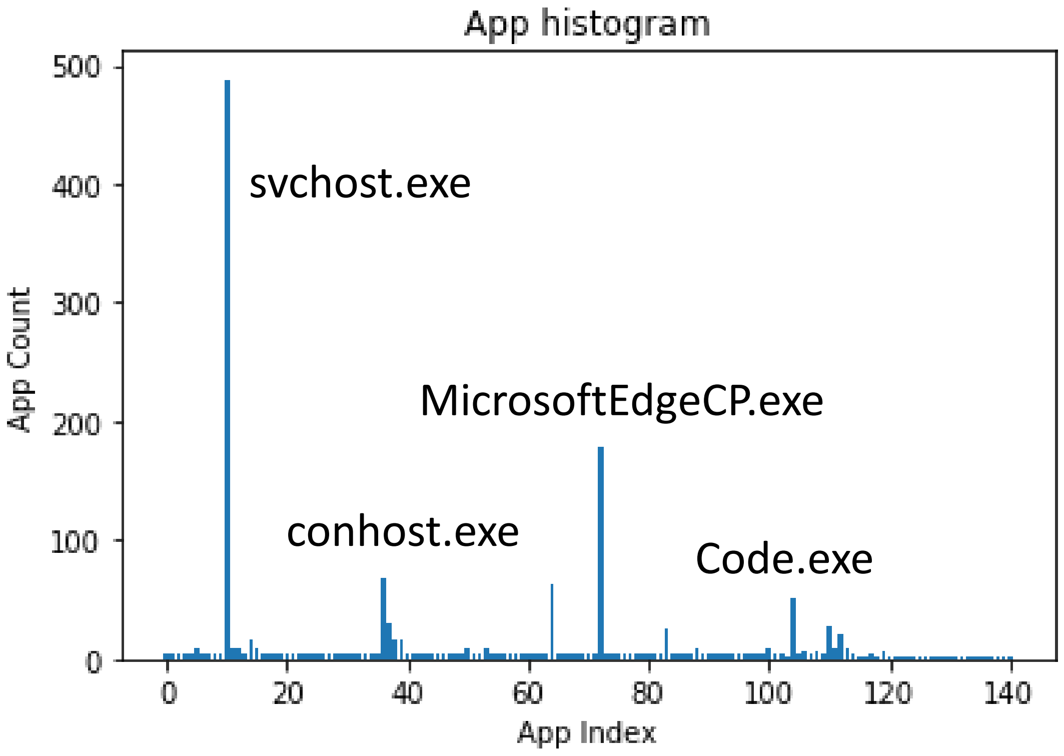
  tokenized link:
  ['google', 'search', 'test', 'test', 'aqs', 'chrome', 'sourceid', 'chrome', 'UTF']



Data Exploration

App data

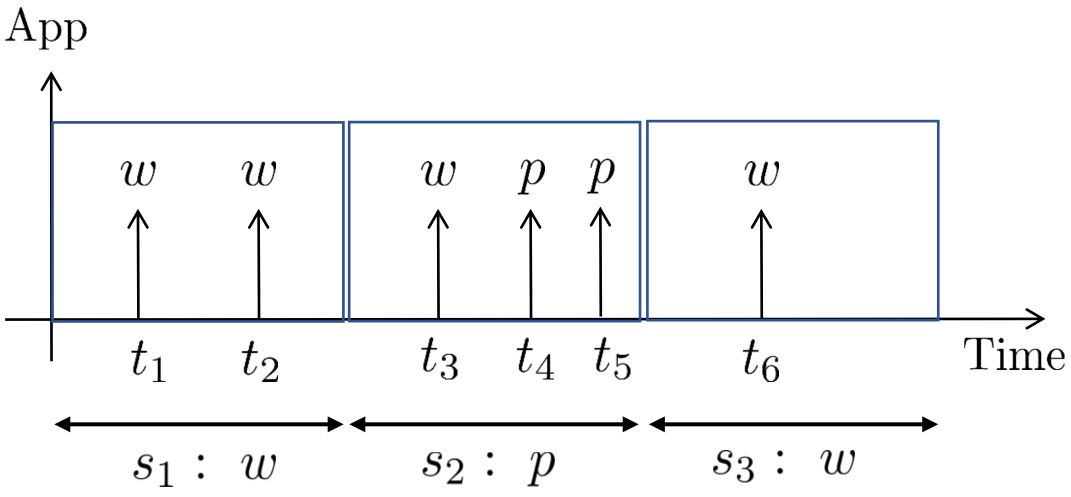
- 1.8M rows, 20K Apps
- Based on App Interactivity dataset from cosmos database (commercial vs non-commercial apps)
- App data is one-hot encoded (same Hamming distance between apps)
- Can be ambiguous: 1355 / 20K Apps are labelled both (w)ork and (p)lay (at different times).



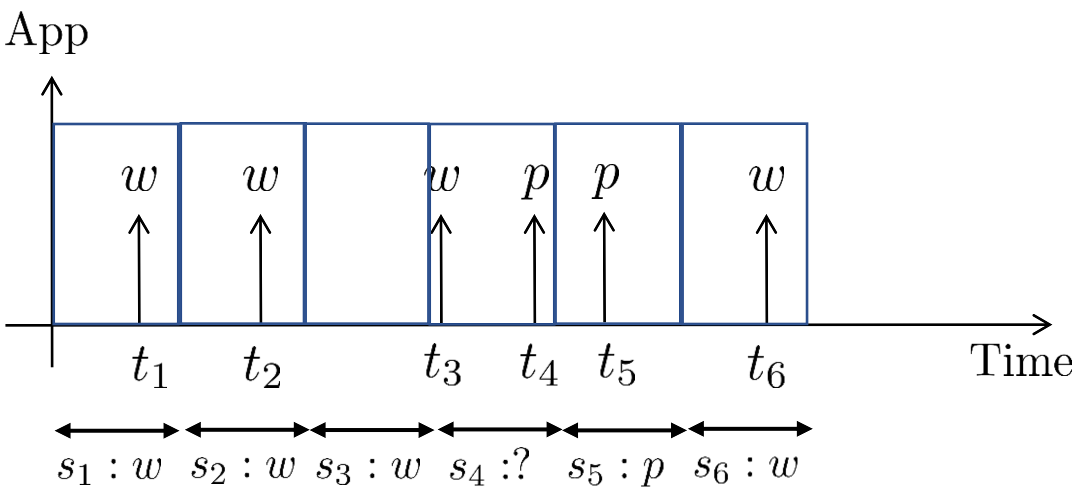
User Sessions

Fixed Session Length

Q1: What is *current* session type and start-time?
Q2: What is expected *next* session type and start-time?



A: Majority Vote

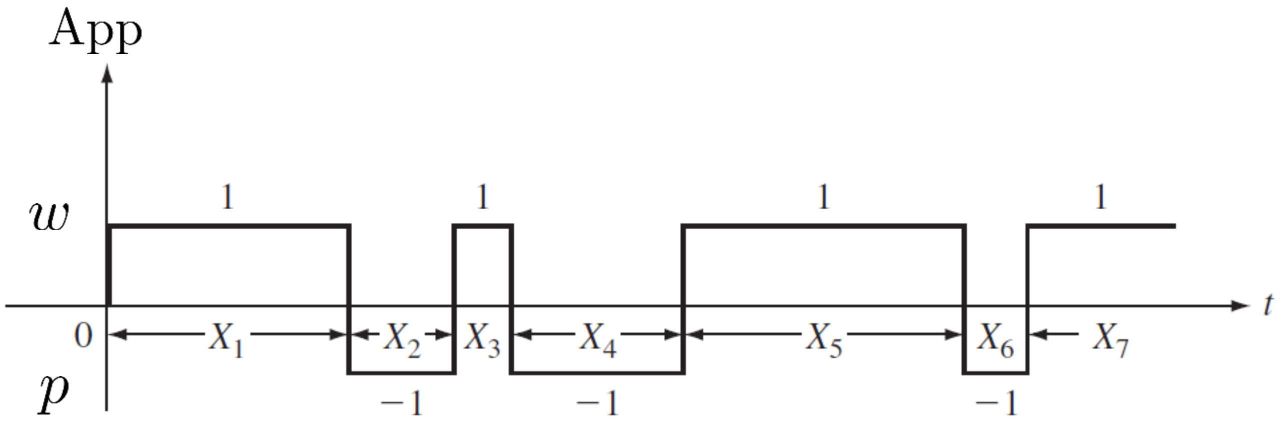


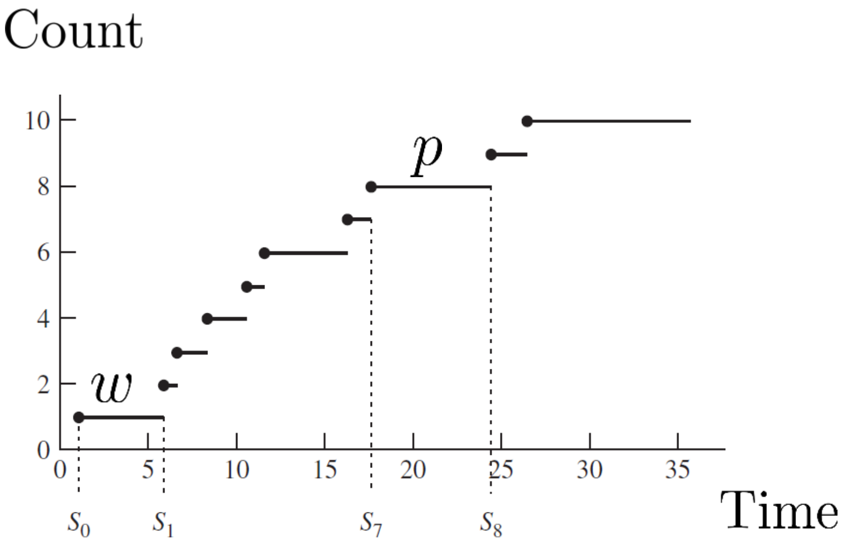
B: Count Based

User Sessions

Variable Session Length

Q: How to predict / learn session duration?





A: Switching Random (Poisson) Process

$$\mathrm{Poiss}(\lambda; k, t) = \frac{(\lambda t)^k}{k!} \exp\{-\lambda t\}$$

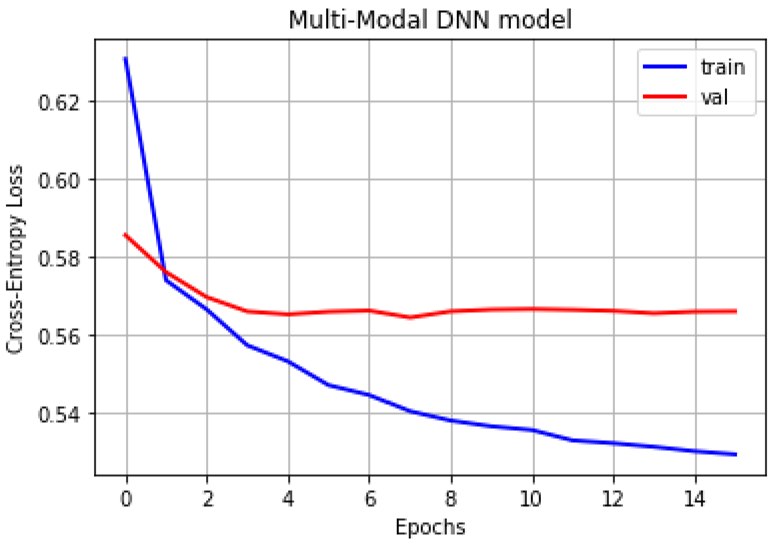
$$X_i \sim \mathrm{Exp}(\lambda) = \lambda \exp\{-\lambda x\}$$

B: Non-homogenous Poisson Process

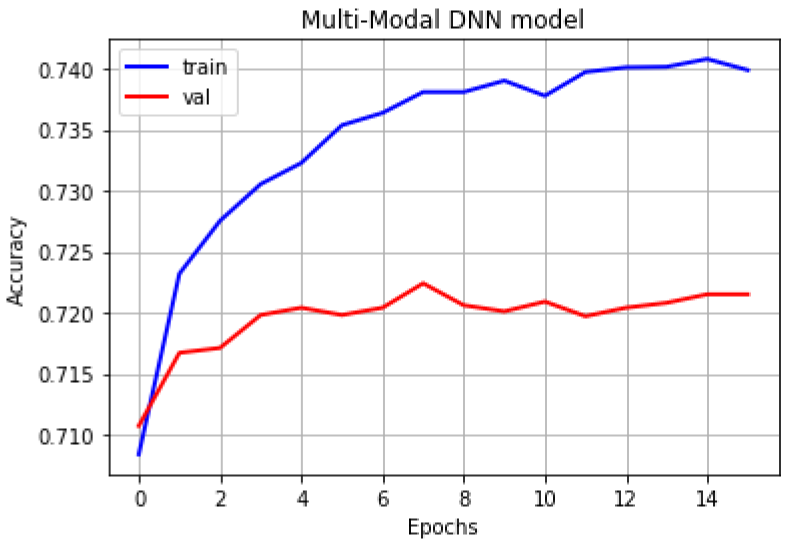
- Rate lambda is a function of time: $\lambda(t)$

Experimental Results

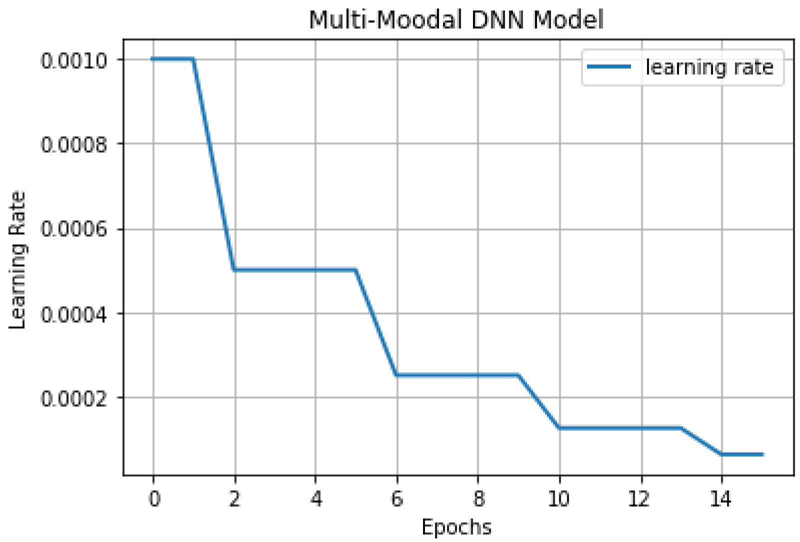
Intent Classification (real dataset)



Flat validation loss = no signs of overfitting



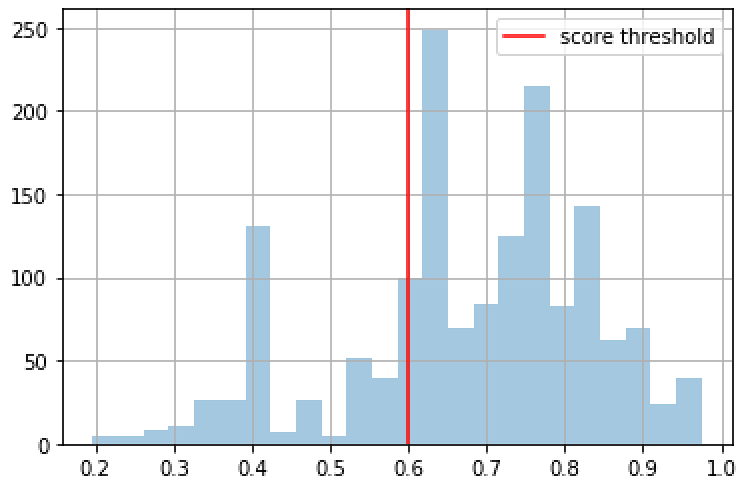
Validation Accuracy is 72%



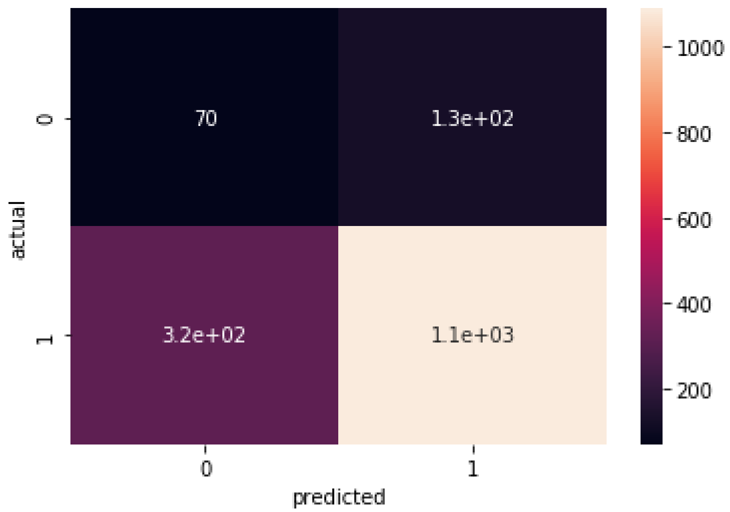
Learning rate schedule divides the rate by 2 every 4 epochs

Experimental Results

Intent Classification (real dataset)



Label = 0 Label = 1



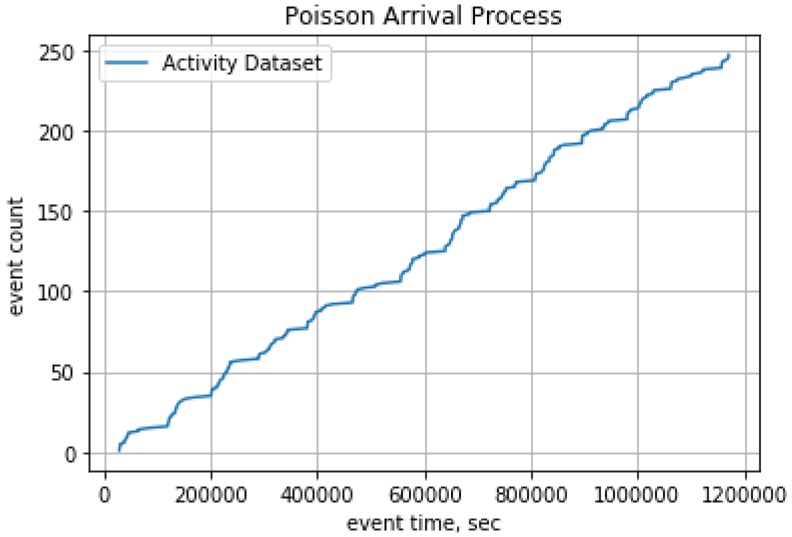
Legend:
0: play
1: work

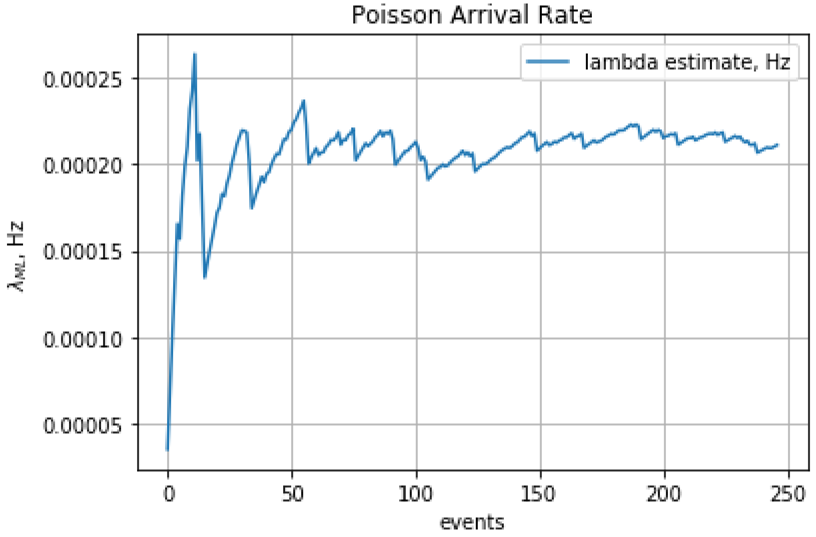
Accuracy:

$$\frac{\mathrm{Tr}(A)}{e^T A e} = \frac{70+1100}{70+130+1100+320} = 0.72$$

Experimental Results

Session Length Prediction:



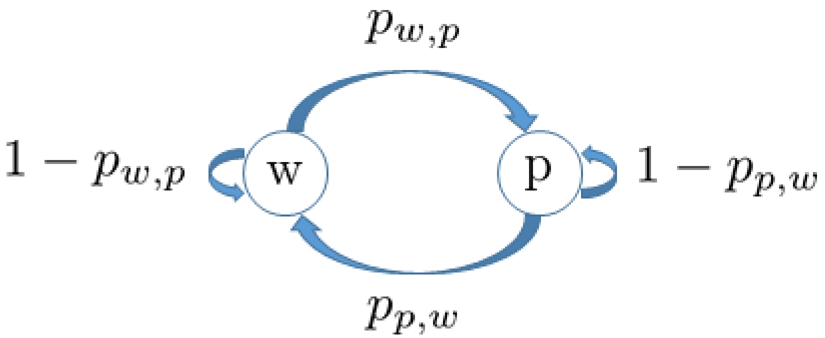


MLE estimate derivation:

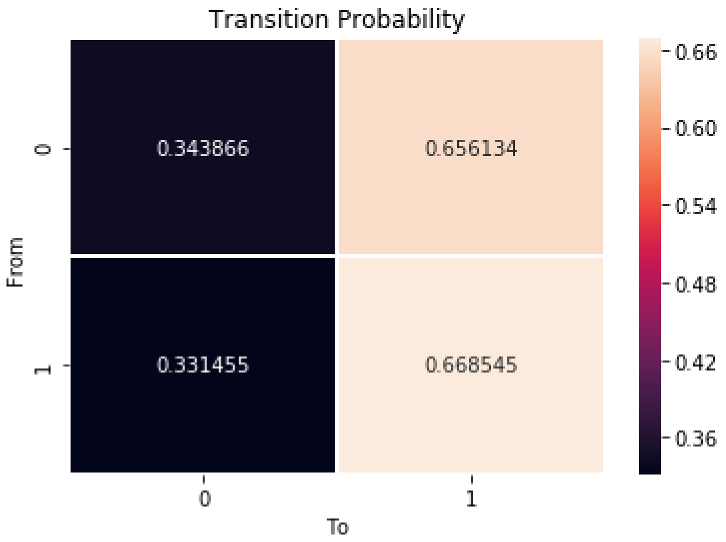
$$
\left[
\begin{aligned}
&p(x_1,\ldots,x_n;\lambda) = \prod_{i=1}^{n} \lambda \exp\{-\lambda x_i\} = \lambda^n \exp\{-\lambda \textstyle\sum_{i=1}^{n} x_i\} \\
&\frac{d}{d\lambda} p(x_1,\ldots,x_n;\lambda) \quad = \quad n\lambda^{n-1} \exp\{-\lambda \sum x_i\} + \lambda^n \exp\{-\lambda \sum x_i\}[-\sum_{i=1}^{n} x_i] = 0 \\
&\hat{\lambda}_{mle} = 1/E[X_i] = 1/\left[\tfrac{1}{n}\textstyle\sum_{i=1}^{n} x_i\right]
\end{aligned}
\right.
$$

Experimental Results

Session Type Prediction:



Markov Chain



$$
\begin{aligned}
\log p(x_1, ..., x_n) &= \log \left[p(x(1) = x_1 \prod_{t=1}^{n} p(x(t+1) = x_t | x(t) = x_t) \right] \\
&= \log q(x_1) + \sum_{t=1}^{n-1} \log p_{x_t, x_{t+1}} \\
&= \log q(x_1) + \sum_{(i,j)} \hat{n}(i,j) \log p(i,j)
\end{aligned}
$$

$\hat{n}(i,j) =$ number of observed transitions from i to j

$\hat{p}_{\mathrm{mle}}(i,j) = \frac{\hat{n}(i,j)}{\sum_k \hat{n}(i,k)}$

Vadim Smolyakov

ABOUT ME

College

Massachusetts Institute of Technology

Computer Science and Artificial Intelligence (CSAIL) Lab

Microsoft Team

Catalyst Team in Enterprise and Security (ENS)

Data Science PhD Intern

Summer in Seattle