HDR-Nets 2020

MoGAN: GAN based Next PoA Selection for Proactive Mobility Management

Boyun Jang

boyunj0226@skku.edu

Dept. of Artificial Intelligence
Sungkyunkwan University, Korea

13th October 2020

Overview

- Problem Statement
- Deep Learning Approaches for Prediction
- MoGAN
- Evaluations
- Conclusions

Problem Statement

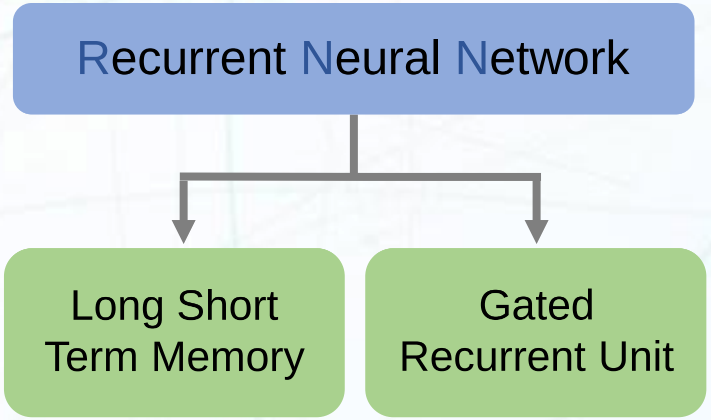
- Requirement
  - Sub 10ms mobility delay for 5G
- Problem
  - User assisted reactive mobility management in 4G is potential bottleneck for 5G
- Solution
  - Proactive mobility management

Problem Statement

- Requirements of proactive mobility management
  - Next Point of Attachment (PoA) prediction with high accuracy
  - Optimal decision for handover trigger time
- Challenges
  - Dense and Ultra-dense cell deployment in 5G
  - Real-time prediction and decision algorithms
- Solution
  - This work focuses on prediction of next PoA
  - A GAN based next PoA prediction mechanism

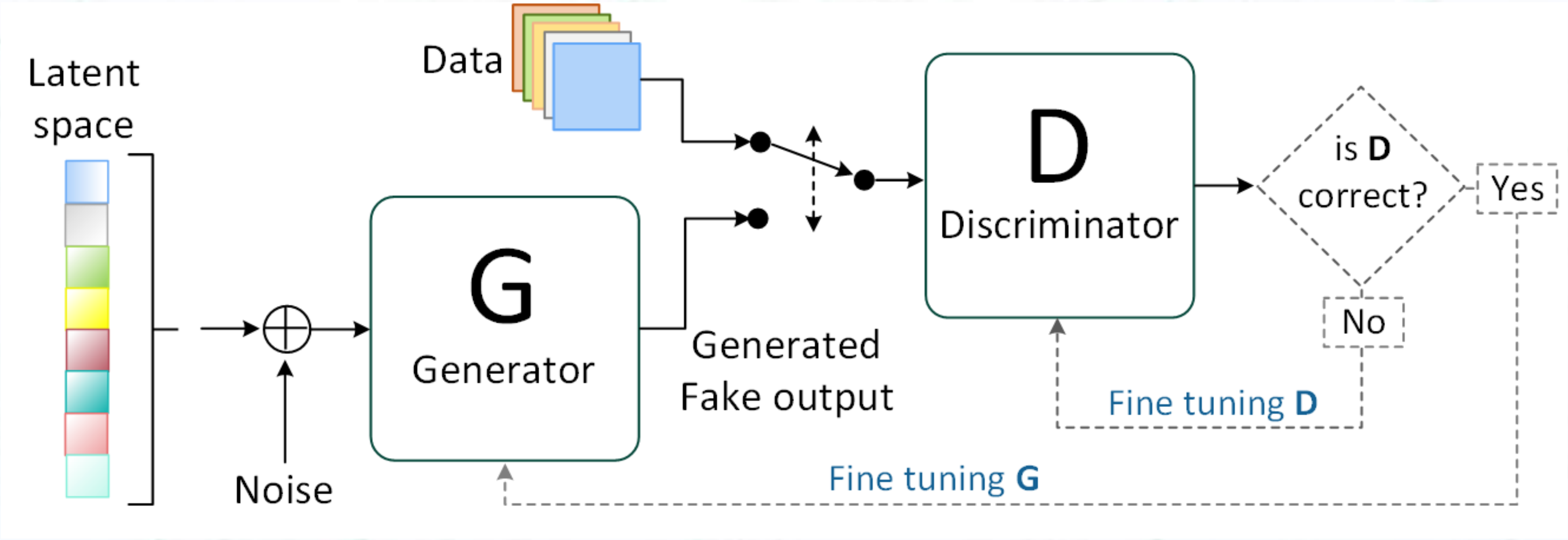
DL Approaches for Prediction

- Recurrent Neural Network (RNN)
  - Pros: Available to capture the feature of continuous data
  - Cons: Gradient vanishing problem occurs with long sequence length
    → Long-term dependency
- Long Short-Term Memory (LSTM)
  - Pros: Additional cell states enable to save more information of past sequences
  - Cons: Complex structure results more computational cost
- Gated Recurrent Unit (GRU)
  - Computationally less expensive
  - Better performance for less complex data



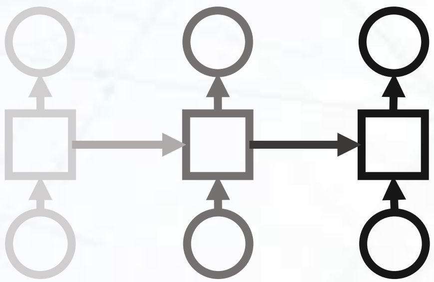
DL Approaches for Prediction

- GAN (Generative Adversarial Network)
  - Various usage →
    - Data generating model
    - Classification model
    - **Prediction model**



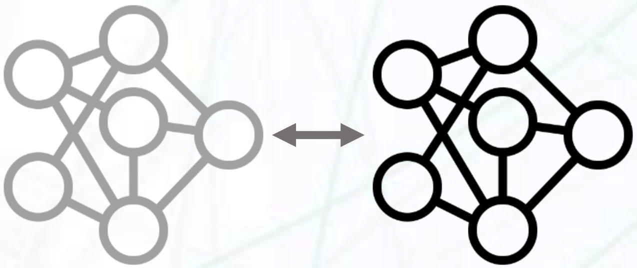
DL Approaches for Prediction

LSTM



- Usually used for prediction
- Large amount of data is required for training
- Former study achieved 91% of accuracy

GAN



- Useful feedback from its adversaries
- Trained to reflect whole distribution of data
- Various types of data are available

MoGAN

- Data preprocessing
  - Movement history (Sequence of PoA)
    - Suppose the actual length of sequence is 5
    - Movement history with less than 5 PoAs: ignored
    - Movement history with more than 5 PoAs: divided

$$\{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$$

$$\{x_1, x_2, x_3, x_4, x_5\}$$

$$\{x_2, x_3, x_4, x_5, x_6\}$$

$$\{x_3, x_4, x_5, x_6, x_7\}$$

  - For each PoA → Transform into One-hot vector
    - N-dimensional vector if there are total N points in the data

MoGAN

- Architecture
  - Model for predicting next PoA of mobile devices
  - Learns from the data consisting of previous sequences

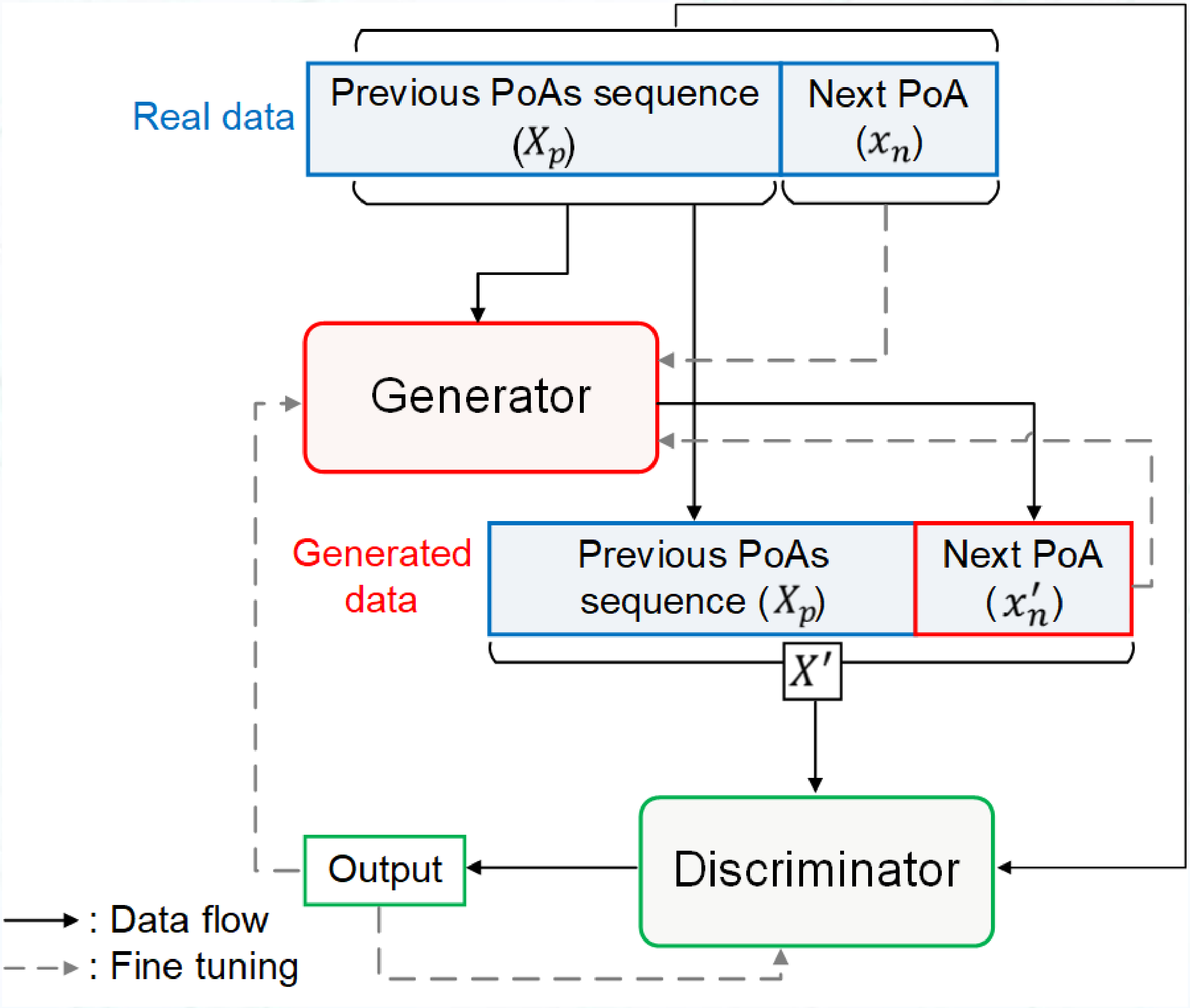
Generator

- Learns the distribution of previous PoA connections
- Generates probable next PoA
- Used as prediction model after training completed

Discriminator

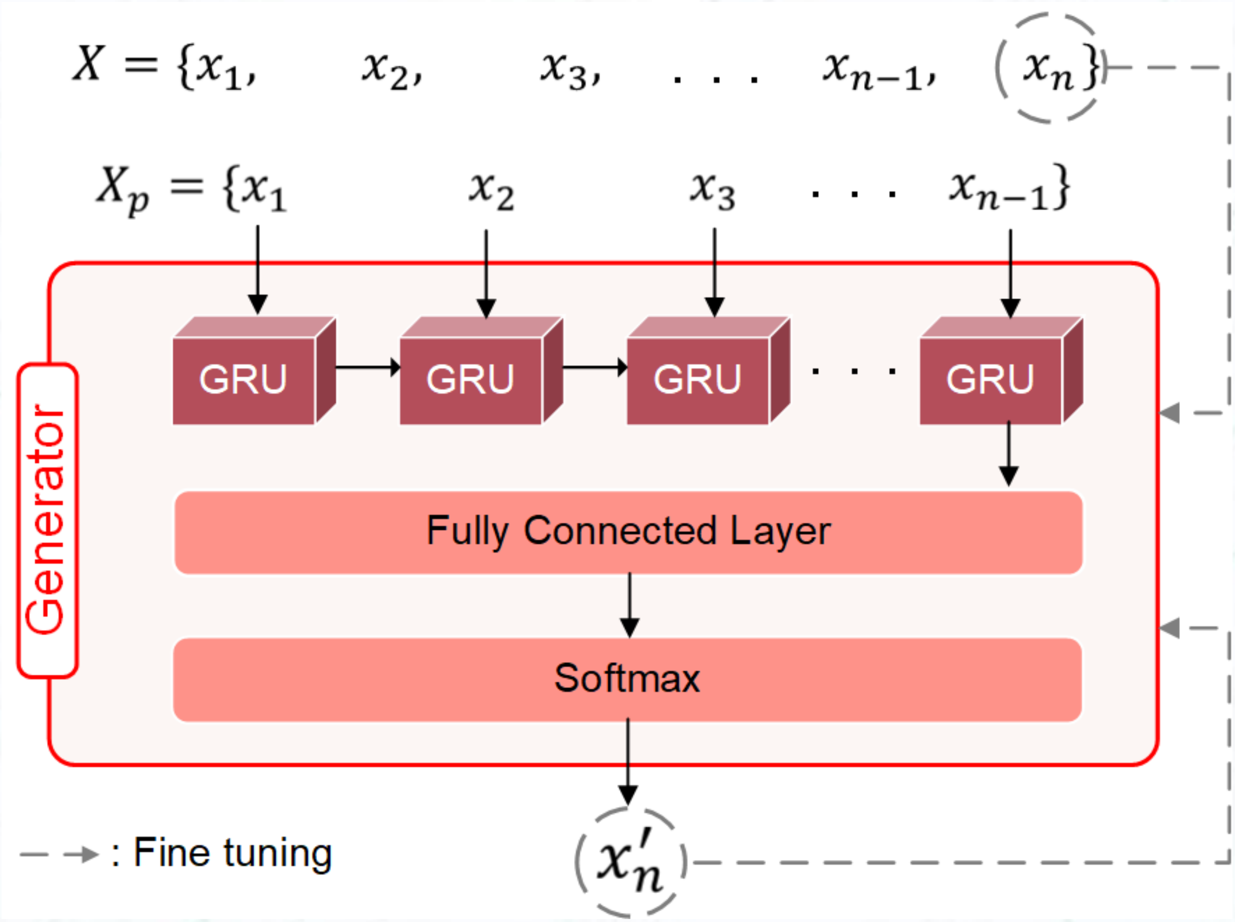
- Classifies between real PoA sequences and generated ones
- Useful feedback for generator

MoGAN


Real data
Previous PoAs sequence
(X_p)
Next PoA
(x_n)
Generator
Generated data
Previous PoAs sequence (X_p)
Next PoA
(x'_n)
X'
Output
Discriminator
: Data flow
: Fine tuning

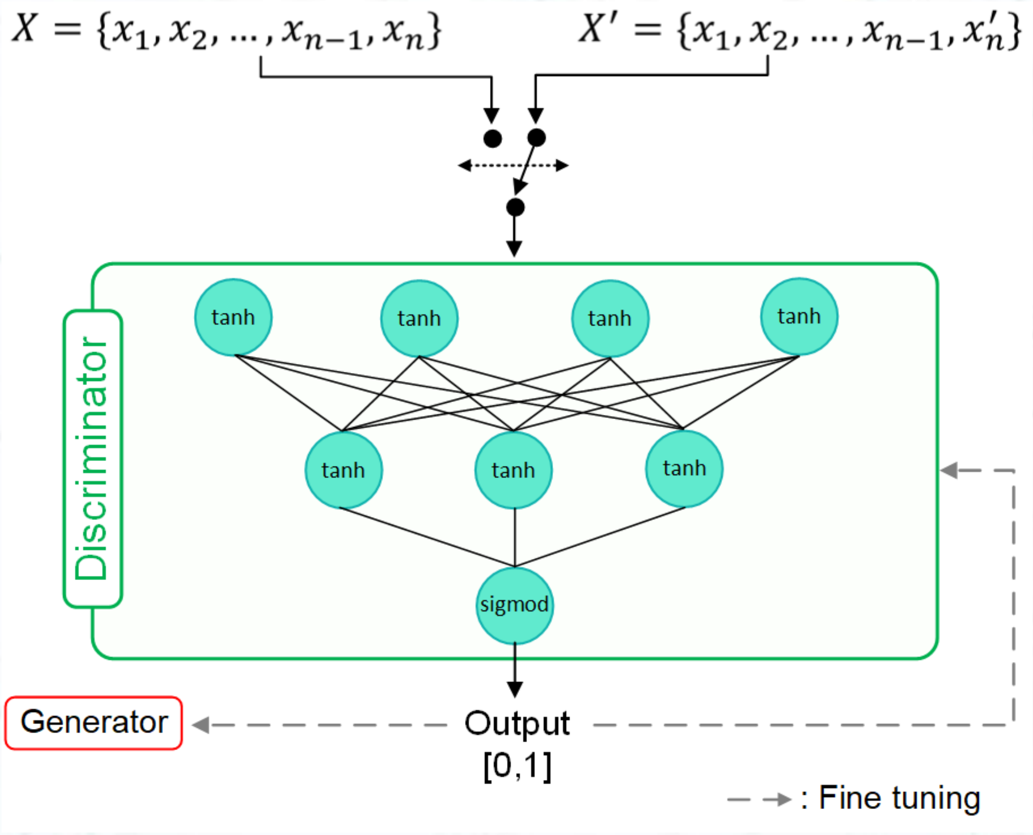
MoGAN

- Structure of generator



MoGAN

- Structure of discriminator
  - For classification, FC layer performs better than RNN based structures
    - For recognizing all the properties of structures



MoGAN

- Training procedure of MoGAN
  - Error function: Binary cross entropy

$$H(y, y') = -\frac{1}{N}\sum_{i=1}^{N}(y_i \log(y_i') + (1 - y_i)\log(1 - y_i'))$$

    - y : Expected value, y': Predicted value

  - Step 1 (Minimax step)
    - $\min_{G_\theta} \max_{D_\phi} [H(0, D_\phi(X)) + H(1, D_\phi(X_p + G_\theta(X_p)))]$
  - Step 2 (Additional training step for generator)
    - $\min_{G_\theta} [H(x_n, G_\theta(X_p))]$

MoGAN

Algorithm: The training procedure of MoGAN

Initialize: Number of total epoch n, number of Step 2 per epoch α,
randomly initialized weights θ, ϕ for G_θ, D_ϕ

Input: $\boldsymbol{X} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_{n-1}}, \boldsymbol{x_n}\}$

1. Error function ← binary cross entropy H
2. for n do:
3. $G_\theta(X_p)$ predicts next PoA ${x_n}'$
4. X' ← Combine X_p with ${x_n}'$
5. D_loss_real ← Get loss value from D for real data $H(0, D_\phi(X))$
6. D_loss_fake ← Get loss value from D for generated data $H(1, D_\phi(X'))$
7. Update ϕ to maximize D_loss_real + D_loss_fake

Continue…

MoGAN

Algorithm: The training procedure of MoGAN

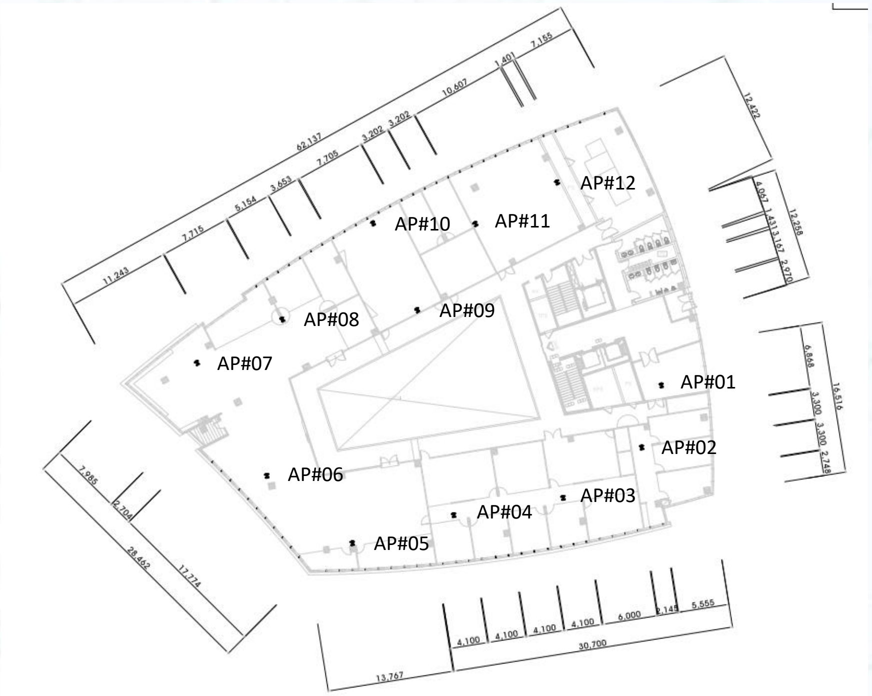
Initialize: Number of total epoch n, number of Step 2 per epoch α,
randomly initialized weights θ, ϕ for G_θ, D_ϕ

Input: $\boldsymbol{X} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_{n-1}}, \boldsymbol{x_n}\}$

8. G_loss_Step1 ← Get loss value from G for $H(1, D_\phi(X_p + G_\theta(X_p)))$
9. Update θ to minimize G_loss_Step1
10. for α do:
11. G_loss_Step2 ← Get loss value from G for $H(x_n, G_\theta(X_p))$
12. Update θ to minimize G_loss_Step2

Evaluations

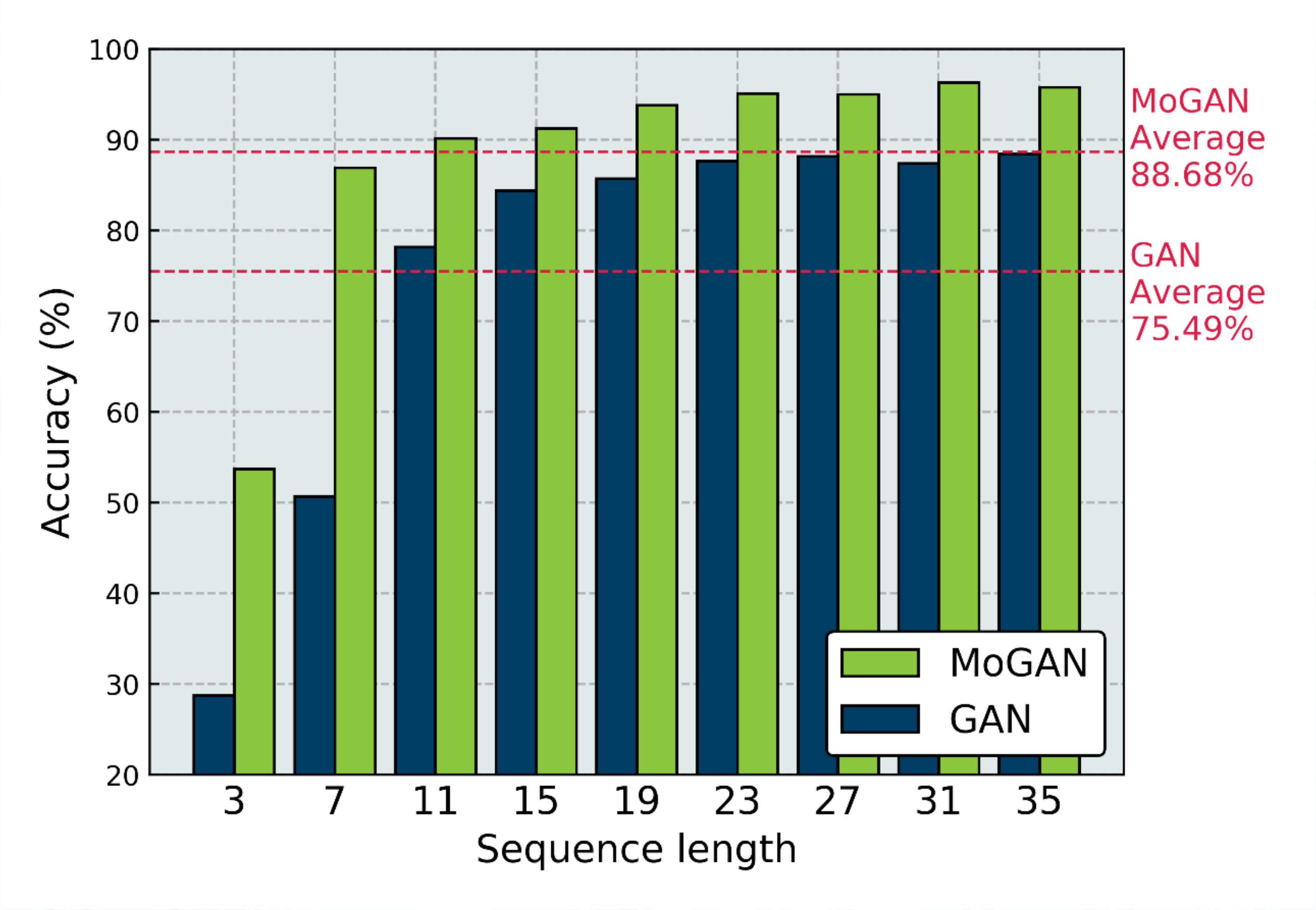
- CMD (Campus Mobility Dataset)
  - Collected from the wireless network of intelligent ICT Convergence Research Center in Pangyo, Republic of Korea
  - 12 APs, 289 users



- Configuration
  - Generator : GRU (512 nodes) + Output layer (12 nodes, softmax)
  - Discriminator : FC (128 nodes, tanh) + FC (64 nodes, tanh)
+ Output layer (1 node, sigmoid)
  - Adam optimizer (lr=0.001), 4000 epochs, 31 sequence lengths, $\alpha = 1$
  - Data → Training : Test = 7 : 3

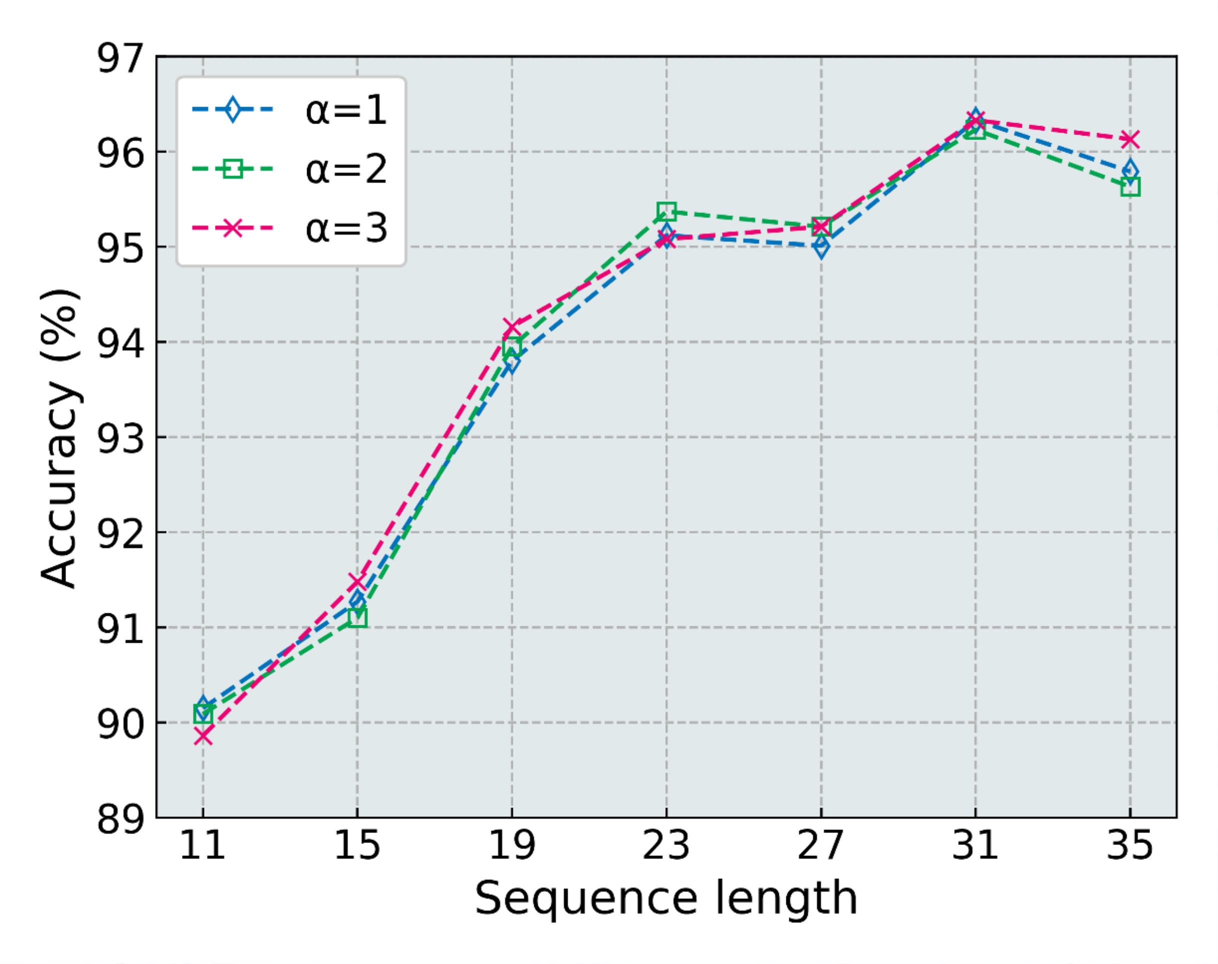
Evaluations

- Next PoA prediction accuracy comparison between MoGAN and vanilla GAN for different sequence lengths



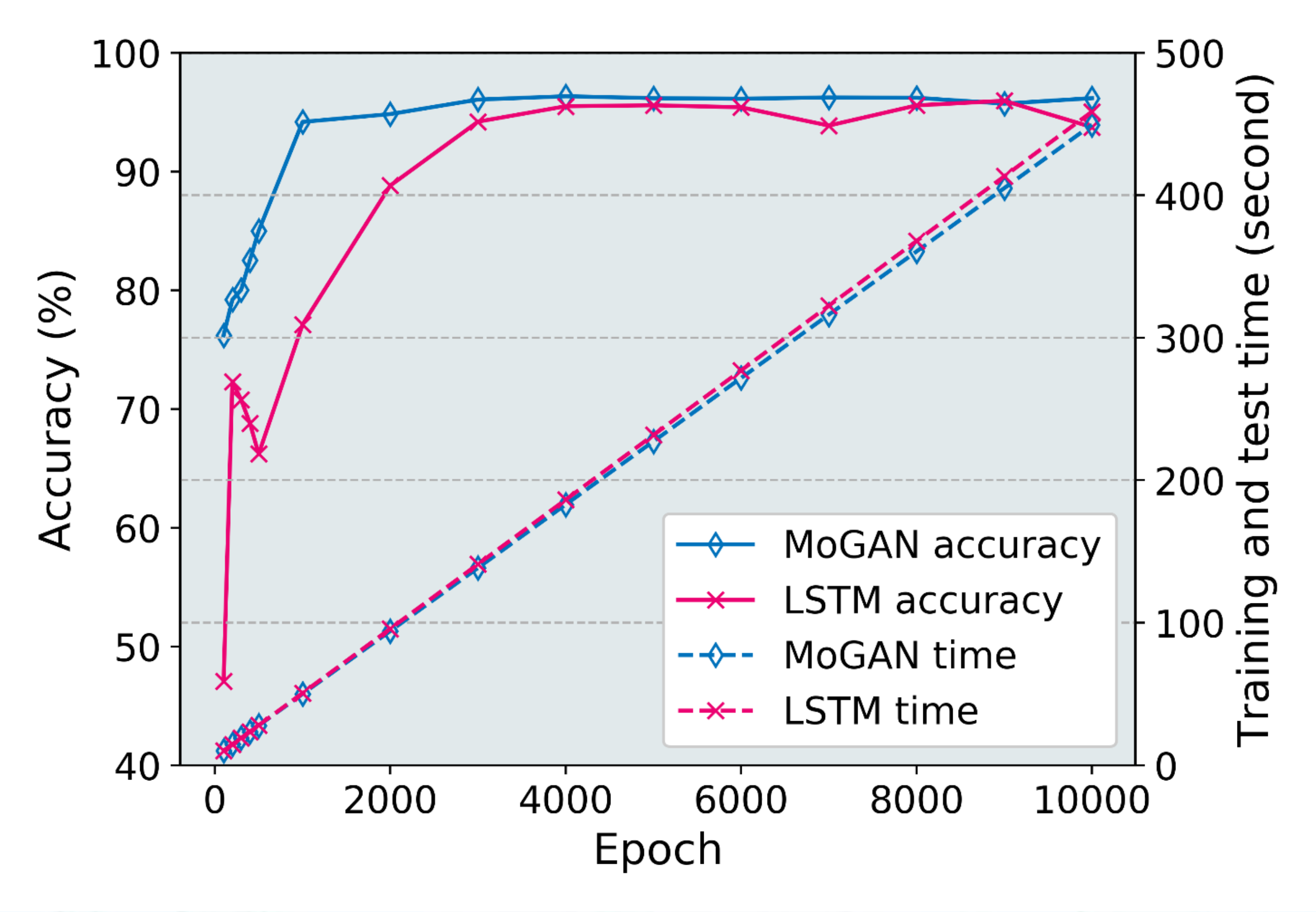
Evaluations

- Analysis of MoGAN with different iterations of Step 2 training (α) with increasing sequence length



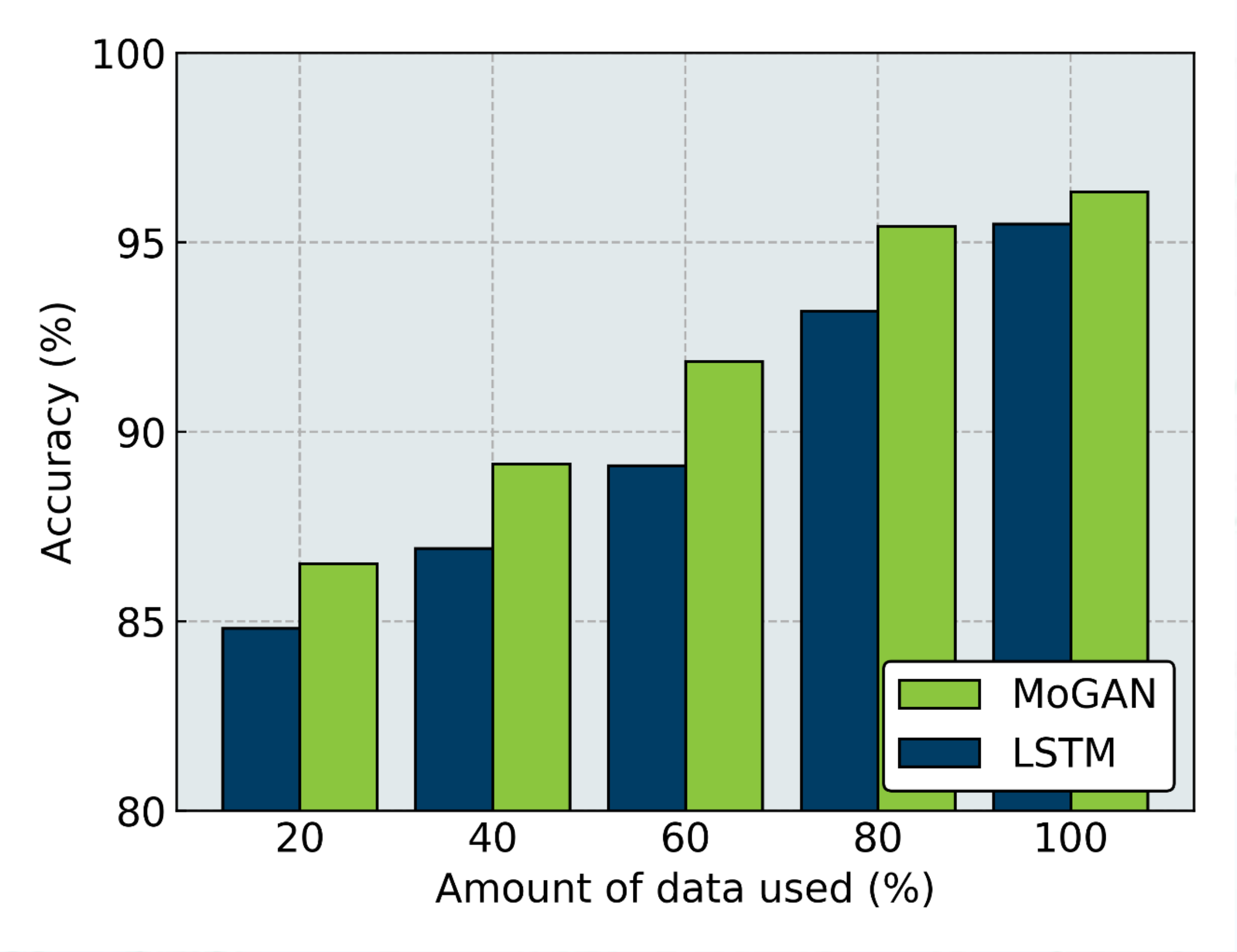
Evaluations

- Performance comparison between MoGAN and stacked LSTM in terms of next PoA prediction accuracy and time cost



Evaluations

- MoGAN and LSTM performance comparison with limited data



Conclusions

- MoGAN achieved maximum 96.33% accuracy
  - For perspective this means that if 3,000 users perform handover at a given time, MoGAN correctly predicts next PoA for 2,890 users
  - Predicting next PoA for a user takes 5.85ms, which makes MoGAN suitable to be used in real mobile network
- Improved method for data-based prediction is suggested which can be used in other domains
- Future work
  - Improvement of MoGAN through other attention mechanisms
  - Extend MoGAN from single step to multiple step prediction

Thank You!