Report Date: 07/15/2022

To: ematson@purdue.edu, ahsmith@purdue.edu, lhiday@purdue.edu and lee3450@purdue.edu
From: BEST (Beacon-based Evacuation System and Technology)
Bacon Beacon

- Hwawon Lee (andylhw12@soongsil.ac.kr)
- Yoonha Bahng (tlol91@cau.ac.kr)
- Dohyun Chung (sosvast@cau.ac.kr)
- Jiwon Lim (senta2006@kw.ac.kr)
- Suhyun Park (2061013@pcu.ac.kr)
- Seongmin Kim (aliveksm@kangwon.ac.kr)
- Myoung Oh (oh278@purdue.edu)

Summary

Q-Learning Algorithm successfully implemented on Server, field test on iOS succeeded. However, to improve the accuracy of the real-world, a number of targets of beacons to classification needed. Implementation of the Flask-SocketIO library was tested, and still working on it. Working on the Design and Experiment section of the article started. Collecting real data of Neural Network models needed.

What "BEST" completed this week

- Tensorflow lite successfully ported on iOS and Android Device[1] [2], working fairly good
  - 99% of accuracy when using Gradient Boost Model
  - 90% of accuracy when using Neural Network
  - However, there were problems at porting tensorflow to tensorflow lite for mobile devices [3]
- Flask-SocketIO implemented [4]
  - Rather than using a multiprocess model, Flask-socketIO can handle both of them.
  - A problem occurred when communicating between Flask-socketIO and swift-socketIO.
- Design and Experiments part of Article started
  - Start drafts on the design and experiment part of article

Things to do by next week

- Augmented Reality at iOS and Android devices need to be started [5]
  - Need study for Augmented Reality, and start thinking about Reality OS and implementation of Apple documents

- Collection of real world data needed
  - As implementation about neural network models finished, need to collect real world data for the evaluation part of the article

- Design and evaluation draft needed to be done
  - As many functions of our project work, draft can be written. Need to organize what we have done, and write the rest of the article by using that data

Problems or challenges:

- Tensorflow lite was not successfully ported
  - RSSI of 21 beacons were used for training phase, and testing phase. Collect it for 5 minutes for every 30 cells.
  - 99% accuracy was shown up, which was not as good for real world data because of overfitting problem
  - Asked Prof. Anthony for solution about our problem, and he said that too much beacons were used for localization
  - Collecting RSSI of 5 strong signal beacons cause accuracy of 90%, and classification in real world works better than collecting all data [11] [12]

- Divided the area of K-SW building for indoor localization and evacuation algorithms
  - Tended to divide into 2.75m x 2.75m for K-SW building, but after measuring the real length of the building, concluded that it does not fit when we divide into that.
  - Divide the area with a flexible rectangle, and train with that data

- Q-learning algorithm does not works well as thought
  - The main reason was when calculate every cells for start, the Q-Table value changes that user cannot get the effective route to escape
  - Not using an individual q-table for the solution of that problem. However, the time takes too long when slicing the area into 2.75m x 2.75m square
  - By adopting a graph based on the area mentioned at Problems or challenges, it takes below 2 seconds to suggest the evacuation route [3]

- Q-learning algorithm is not as fast as Dijkstra algorithm
  - When collecting data for evaluation part, noticed that Dijkstra algorithm is 20 times faster than Q-Learning algorithm
  - Asked Dr. Matson for a solution, and heard the reason to use reinforced learning for escape algorithms
  - Not only suggesting the shortest path, BEST now suggests different paths to make people evacuate without congestion

References

[1] Tensorflow, “Tensorflow Lite”, Tensorflow, “https://www.tensorflow.org/lite/guide” (accessed Jul. 15, 2022)

[2] Tensorflow, “Convert TensorFlow models”, Tensorflow, “https://www.tensorflow.org/lite/models/convert/convert_models” (accessed Jul. 15, 2022)

[3] Github, “Regression: TFLiteConverter changes output order again since 2.7”, Github, “https://github.com/tensorflow/tensorflow/issues/55242” (accessed Jul. 15, 2022)

[4] Miguel Grinberg, “API Reference”, Flask-SocketIO, “https://flask-socketio.readthedocs.io/en/latest/api.html” (accessed Jul. 15, 2022)

[5] Apple, “Apple Developer Documentation”, Apple.com, https://developer.apple.com/documentation/ (accessed Jul. 15, 2022).

[6] Longquan Jiang, Bo Zhang, Qin Ni, Xuan Sun, and Pingping Dong, “ Prediction of SNP sequences via Gini impurity based gradient boosting method,” IEEE ACCESS, vol. 7, pp. 12647-12657, 2019.

[7] A. Kumar, “building-evacuation-q-learning,” GitHub, Jun. 07, 2022. https://github.com/KumarUniverse/building-evacuation-q-learning (accessed Jun. 10, 2022).

[8] Tensorflow, “Decision Forest”, Tensorflow, “https://www.tensorflow.org/decision_forests/tutorials/automatic_tuning_colab” (accessed Jul. 15, 2022).

[9] Tensorflow, “Automated hyper-parameter tuning ： tensorflow decision forests,” TensorFlow. https://www.tensorflow.org/decision_forests/tutorials/automatic_tuning_colab. (accessed Jul. 15, 2022).

[10] Tensorflow, “GradientBoostedTreesModel”, Tensorflow, “https://www.tensorflow.org/decision_forests/api_docs/python/tfdf/keras/GradientBoostedTreesModel” (accessed Jul. 15, 2022)

[11] Tensorflow, “Model Optimization”, Tensorflow, “https://www.tensorflow.org/lite/performance/model_optimization” (accessed Jul. 15, 2022)

[12] Tensorflow, “Introduction to Keras Tuner”, Tensorflow, “https://www.tensorflow.org/tutorials/keras/keras_tuner” (accessed Jul. 15, 2022)