# Summer Reading 4: A temporal CNN with a novel convolution operator for human trajectory prediction

Social Robot Navigation Project @ Bot Intelligence Group

Paper:

Zhao, Dapeng, and Jean Oh. "Noticing motion patterns: A temporal cnn with a novel convolution operator for human trajectory prediction." IEEE Robotics and Automation Letters 6.2 (2020): 628-634.
https://ieeexplore.ieee.org/abstract/document/9309403

# Summary:

## Abstract

As more robots share the same space with humans, it is important for robots to predict humans' trajectories for safer navigation. The authors propose Social-PEC (Social Pattern Extraction Convolution). **Social-PEC is a CNN (Convolutional Neural Network) based** approach to learn, detect, and extract patterns in sequential trajectory data.

## Introduction

The authors focus on the pedestrian trajectory prediction in crowded environments. Predicting pedestrian behavior is challenging because not only the physical properties (energy, momentum, etc.) of pedestrians affect but also pedestrians' hidden objectives and subtle social norms in crowd interactions affect their future trajectories.

The existing works used the past trajectories to capture the context of the crowd movements. However, **the authors propose Social-PEC is where trajectories are represented as a combination of motion patterns**.

The authors design their model to understand and learn various pedestrians' motion patterns in crowds, and force the model to notice the motion patterns during training and predict upon them.

The authors build their sequence encoder based on Temporal CNN. They propose a new convolution operator that enables the model to actually detect, learn, and extract motion patterns from the observed trajectories.

## Related Work

Some researches argued that hand-crafting models, Social Force, Interactive Gaussian Process, etc., have various limitations. These led to data-driven techniques that allow the machines to learn directly from data.

***RNN and LSTM, a variation of RNN, have been popularly used to model sequential data.*** However, since RNN tends to gradually forget old information, LSTM became a remedy to selectively forget or remember data. LSTM is a reasonable choice for arbitrarily and extensively long sequences. ***However, in the pedestrian trajectory prediction, excessively long sequences are not necessary since too far past histories of the pedestrians are no longer meaningful to predict their future trajectory.***

Other shortcomings of RNN/LSTM are: vanishing/exploding gradients, unstable and expensive training, and inefficient parameters

Since the hidden space of RNN/LSTM is not well understood and difficult to understand its semantic meanings in a physical space, using a pooling layer (to pick necessary and important data) after RNN/LSTM during information aggregation is limited.

The graph representation is a popular design choice, but the size of the graph depends on the number of pedestrians, so there are scalability challenges.

**The authors propose the Social-PEC model which avoids the issues pointed out in the existing approaches mentioned above.**

1. **Social-PEC takes a fixed length of history trajectory as inputs and does not rely on RNNs to encode trajectory, thus avoiding the RNN training issue.**
2. **The trajectory encoder is based on motion patterns that are intuitively reasonable and can be easily visualized, thus avoiding the obscurity during "social pooling" which is to aggregate the encoded embedding of the neighboring pedestrians' trajectories.**

## Trajectory Prediction with PEC

Humans react to general motion patterns instead of precise location coordinates.

A motion pattern (a sequence of location coordinates) is a short segment of trajectory that can be frequently seen in real trajectory data.

A trajectory is a series of states at a constant time interval. Each state is a 2D location coordinate. In the future, states can include additional information like orientation or personality of pedestrians.

PEC is a mechanism that detects and recognizes patterns from data.

## Evaluation and Discussion

The authors evaluated their model using publicly available 2 datasets, ETH and UCY.

The authors computed the prediction error using 2 different error metrics:

1. Average Displacement Error (ADE) to compute mean euclidean distance over all estimated points at each time step in the predicted and true trajectories
2. Final Displacement Error (FDE) to compute the mean euclidean distance between the final predicted location and the final true location

## Conclusion

The authors used a CNN based model, PEC, and applied it to human trajectory prediction. As PEC is applied to human trajectory prediction, the authors called their model "Social-PEC". The Social-PEC outperformed many existing RNN based models (S-LSTM, SGAN, etc.) that predict human trajectories. The authors propose that CNNs might be a better option for modeling pedestrian trajectories than RNNs.

# Glossary:

CNN (Convolutional Neural Network)

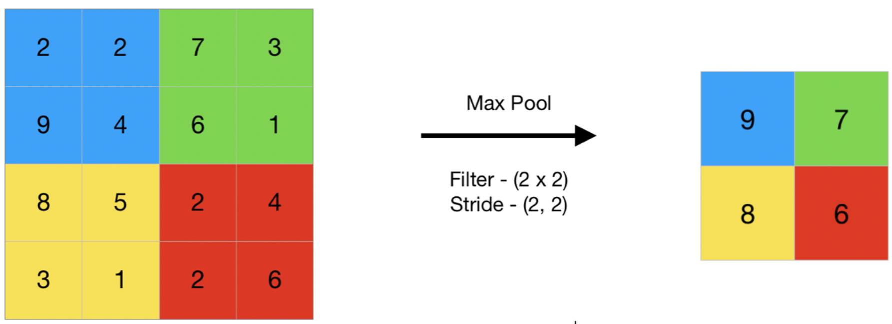
- A Convolutional neural network (CNN) is a neural network that has one or more convolutional layers and are used mainly for image processing, classification, segmentation and also for other auto correlated data.

Sequential data

- When the points in the dataset are dependent on the other points in the dataset, the data is termed sequential.

Pooling layer

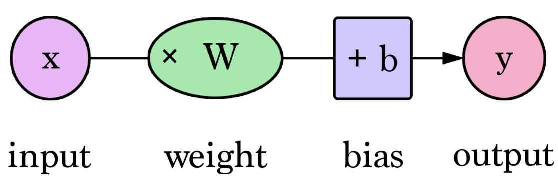
- Pooling layers are used to reduce the dimensions of the feature maps (results from convolutional layers) while keeping necessary and important information/data. Thus, it reduces the number of parameters to learn and the amount of computation performed in the network.




https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/

Weights

- Weight is the parameter within a neural network that transforms input data within the network's hidden layers. A neural network is a series of nodes, or neurons. Within each node is a set of inputs, weight, and a bias value. As an input enters the node, it gets multiplied by a weight value and the resulting output is either observed, or passed to the next layer in the neural network. Y = x*W+b. Often the weights of a neural network are contained within the hidden layers of the network.




https://deepai.org/machine-learning-glossary-and-terms/weight-artificial-neural-network

Loss

- Prediction error of Neural Network.

Backpropagation

- Back-propagation is the essence of neural net training. It is the practice of fine-tuning the weights of a neural net based on the error rate (i.e. loss) obtained in the previous epoch (i.e. iteration).
https://towardsdatascience.com/how-does-back-propagation-in-artificial-neural-networks-work-c7cad873ea7

- After each forward pass through a network, backpropagation performs a backward pass while adjusting the model's parameters (weights and biases).
https://towardsdatascience.com/understanding-backpropagation-algorithm-7bb3aa2f95fd