Dynamic cyclone wind-intensity prediction using co-evolutionary multi-task learning

Dr. Rohitash Chandra

1. Centre for Translational Data Science
http://sydney.edu.au/data-science
2. Discipline of Business Analytics
School of Business

International Conference on Neural Information Processing
Ramada Pearl Hotel, Guangzhou, China
www.iconip2017.org

Overview

Background

- Neuro-evolution refers to the use of evolutionary algorithms for training. Cooperative coevolution (CC) is an evolutionary computation method that decomposes a problem into subcomponents and employs standard evolutionary algorithms in order to gradually solve the bigger problem. The subcomponents are also known as species or modules and are represented as subpopulations.
- The subpopulations are evolved separately and the co-operation only takes place for fitness evaluation for the respective individuals in each subpopulation.

Background and Motivation

- Modular neural networks are motivated from repeating structures in nature. The goal is to store knowledge as modules so that disruption to certain modules do not disrupt the entire network.
- Although neuro-evolution has been successfully applied for training neural networks, multi-task learning for enhancing neuro-evolution has not been fully explored.
- In prediction for climate extremes, it is important to develop models that can make predictions dynamically, i.e robust "on the fly" prediction given minimal information about the event. In time series prediction, it is typical to reconstruct the time series into a state-space vector defined by fixed embedding dimension and time lag.

Background and Motivation

- The embedding dimension defines the minimal timespan or number of past data-points needed to make a prediction. This can be seen as a limitation since data from the past cyclones mostly considered readings every 6 hours. Therefore, the model should have the feature to make timely prediction as soon as the cyclone has been detected.
- Multi-task learning employs shared representation knowledge for learning multiple instances from the same problem. In the case of time series, multi-task learning can consider different a set of embedding dimensions as tasks that have shared knowledge representation.
- There has not been any investigation that explores the embedding dimension of a time series as subtasks for multi-task learning which can be beneficial for dynamic time series prediction.

Overview

- Co-evolutionary multi-tasking learning algorithm is proposed that provides a synergy between multi-task learning and co-evolutionary algorithms.
- The method enables neural networks to feature shared knowledge representation while retaining modularity for robust "on the fly" prediction considered to be dynamic time series.
- The original time series is reconstructed with a set of values for the embedding dimension that defines the subtasks for multi-task learning. Here, the tasks are referred as subtasks to avoid confusion with typical multi-task learning where a set of datasets define the tasks.
- The proposed method is used for tropical cyclone wind-intensity prediction and addresses the problem of dynamic prediction.

Time series prediction: State-space reconstruction

- In state-space reconstruction, the original time series is divided using overlapping windows at regular intervals that can be used for one-step-ahead prediction. Hence, given an observed time series $x(t)$, an embedded phase space $Y(t) = [(x(t), x(t-T), ..., x(t-(D-1)T)]$ can be generated, where, T is the time delay, D is the embedding dimension (window), $t = (D-1)T, DT, ..., N-1$, and N is the length of the original time series.
- The optimal values for D and T must be chosen in order to efficiently apply Taken's theorem

Proposed Problem: Dynamic time series prediction

- Dynamic time series prediction refers to the ability of a model to make prediction for different number of input features or set of values of the embedding dimension. Therefore, one step ahead prediction $\hat{y}_{t+1}$ for subtask Ω_m that feature the respective embedding dimension can be given by

$$\begin{aligned} \Omega_m &= x_t, x_{t-1}, ..., x_{t-m} \\ \hat{y}_{t+1} &= f(\Omega_m) \end{aligned} \tag{1}$$

where $f(.)$ is a model such as a feedforward neural network, m is defines the length of the input features, where $m = 1, 2, ..., M$, and M is the number of subtasks.

Proposed Method: Co-evolutionary multi-task learning

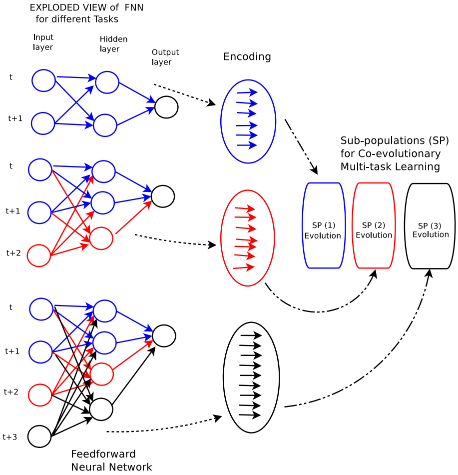

Figure: Subtasks in co-evolutionary multi-task learning. Subtask 1 employs a network topology with 2 hidden neurons while the rest of the subtasks add extra input and hidden neurons.

Co-evolutionary multi-task learning

The base module is given as $\Phi_1 = [\omega_1, \upsilon_1]$. Multi-task learning is used via coevolution to update the respective network module Φ_m given subtask Ω_m. Note that the cascaded network module θ_m of subtask m is constructed by combining with current Φ_m and previous network module Φ_{m-1} as follows.

$$
\begin{aligned}
\Phi_1 &= [\omega_1, \upsilon_1]; \;\; \theta_1 = (\Phi_1) \\
\Phi_2 &= [\omega_2, \upsilon_2]; \;\; \theta_2 = [\theta_1, \Phi_2] \\
&\vdots \\
\Phi_M &= [\omega_M, \upsilon_M]; \;\; \theta_M = [\theta_{M-1}, \Phi_M]
\end{aligned} \tag{2}
$$

Co-evolutionary multi-task learning

The list of network modules considered for training or optimisation is therefore $\mathbf{\Phi} = (\Phi_1, \ldots, \Phi_M)$.

$$
\begin{aligned}
y_1 &= f(\theta_1, \Omega_1) \\
y_2 &= f(\theta_2, \Omega_2) \\
&\vdots \\
y_M &= f(\theta_M, \Omega_M)
\end{aligned}
\tag{3}
$$

Algorithm - CMTL

Alg. 1 Co-evolutionary multi-task learning

```
Data: Reconstructed state-space vector for the respective subtasks Ω_m.
Result: Prediction error for the respective subtasks Ω_m.
initialisation
for each subtask Ω_m do
    1. Define different sub-populations S_m using network module Φ_m
    2. Initialise all individuals in sub-population S_m
end
while each phase until termination do
    for each sub-population S_m do
        for each generation until depth β do
            for each i individual in sub-population S_m_ij do
                for each j in individual S_m_ij do
                    Assign individual V_m = S_m_ij
                end
                if m == 1 then
                    1. Z_m = [V_m]
                    2. Fitness evaluation via Equation 4 by encoding Z_m in the cascaded network module θ_m
                end
                else
                    1. Append to best Individual B_(m-1) of previous sub-population: Z_m = [V_m, B_(m-1)]
                    2. Fitness evaluation via Equation 4 by encoding Z_m in the cascaded network module θ_m
                end
            end
            for each i individual in sub-population S_m_ij do
                * Select and create new offspring via evolutionary operators:
                1. Selection, 2. Crossover, and 3. Mutation
            end
        end
    end
end
for each subtask Ω_m do
    1. Get best solution B_m from sub-population S_m
    2. Load test data
    3. Report prediction performance given by loss L_m.
end
```

Design of Experiments

- An experimental study that compares the performance of CMTL with conventional evolutionary (single task learning) methods such as cooperative neuro-evolution (CNE) and an evolutionary algorithm (EA) for dynamic time series prediction.
- Tropical cyclones from South Pacific and South Indian Ocean are considered.

Dataset

The original data of tropical cyclone wind intensity in the South Pacific was divided into training and testing set as follows:

- Training Set: Cyclones from 1985 - 2005 (219 Cyclones with 6837 data points)
- Testing Set: Cyclones from 2006 - 2013 (71 Cyclones with 2600 data points)

In the case for South Indian Ocean, the details are as follows:

- Training Set: Cyclones from 1985 - 2001 (285 Cyclones with 9365 data points)
- Testing Set: Cyclones from 2002 - 2013 (190 Cyclones with 8295 data points)

South Indian Ocean

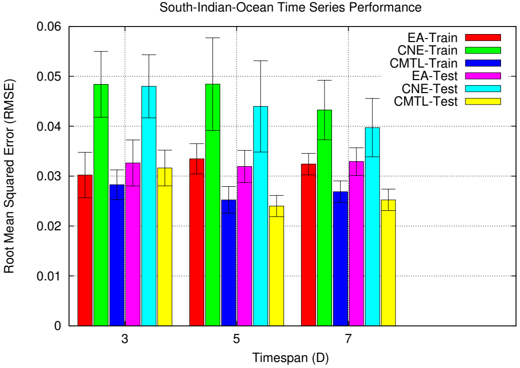

Figure: Performance given by EA, CNE, CMTL for South Indian Ocean

South Pacific Ocean

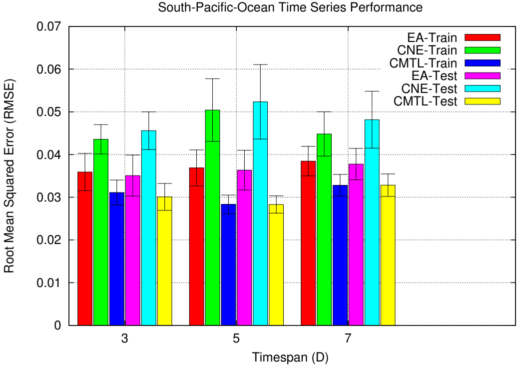

Figure: Performance given by EA, CNE, CMTL for South Pacific Ocean

Discussion

- The goal of the experiments was to evaluate if the proposed algorithm can deliver similar results when compared to single task learning methods for the dynamic time series problems.
- The comparison was to ensure that the approach does not lose quality in terms of generalisation performance when compared to single-task learning methods.
- The results have shown that CMTL not only addresses dynamic time series, but also improves the performance when compared to single-task learning.
- Although feedforward neural networks have been used in CMTL, other neural network architectures, and learning models can be used depending on the nature of the problem.

Conclusions and Future Work

- Co-evolutionary multi-task learning is needed for tropical cyclones where robust prediction needs to be given. The method can work even when a single point is implemented as a subtask. As more points of data are given to the subtask, more predictions can be made which makes the model dynamic and robust.
- The results show that the proposed algorithm not only addressed dynamic time series, but also improved prediction performance when compared to conventional methods.
- In future work, the proposed algorithm can be used for other time series problems that can be broken into multiple subtasks. In case of tropical cyclones, which is a multivariate problem, the different subtasks can be redefined with information such as cyclone tracks, seas surface temperature, and humidity.

Thank You

More information: https://rohitash-chandra.github.io