Learning Graph Convolutional Network for Skeleton-based Human Action Recognition by Neural Searching

Nguyen Anh Tung

Student ID: 2020 – 25375

Seoul National University

CONTENT

PROBLEM

METHOD

EVALUATION

1.

PROBLEM

Let’s start with the problem we need to solve

PROBLEM

- Human action recognition from Skeleton data fueled by the Graph Convolutional Network.
- Most GCN methods ignores implicit joint correlations.
- The need of replacing the fixed graph structure with dynamic one.
- Higher-order connections are not well involved

2.

METHOD

Let’s continue with the proposed method

METHOD

- Using Automatic Neural Architecture Search (NAS)
- Searching in a GCN space built with multiple graph function modules.
- Using Search strategy for both sampling and memory efficient.

NAS

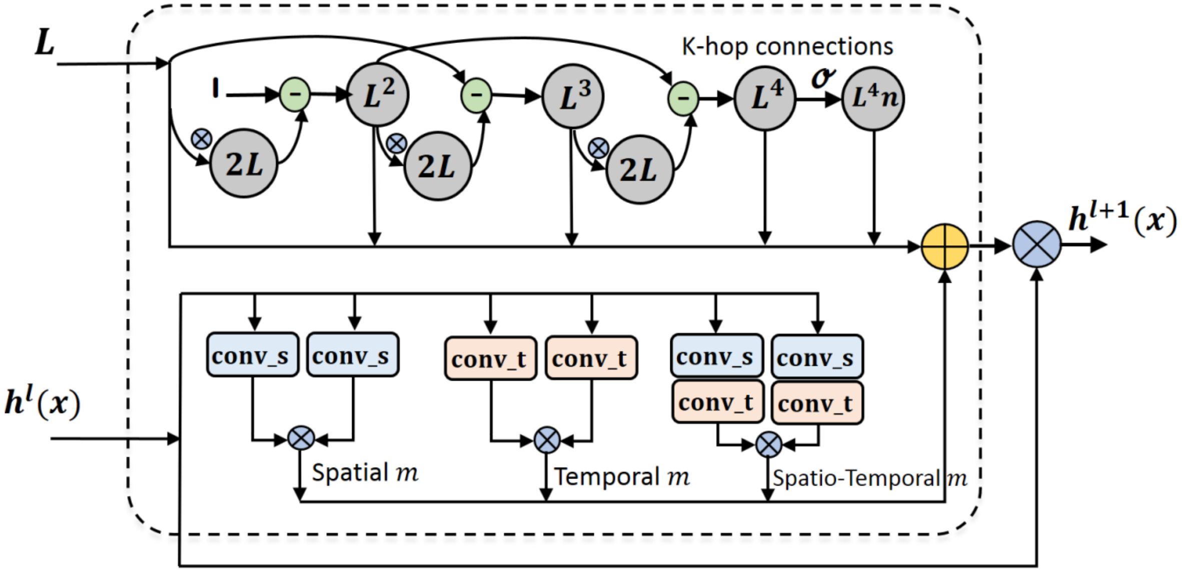
Search
Space
Search
Strategy
Performance
Estimation
Strategy

METHOD

GCN

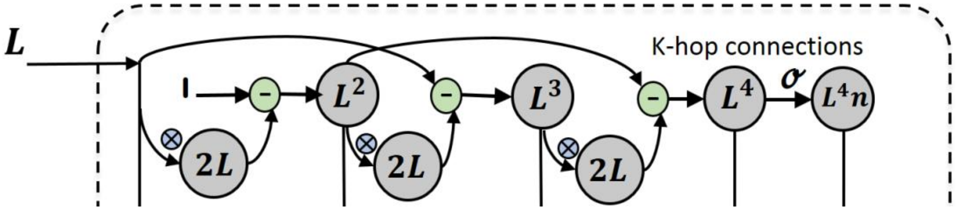
SEARCH

SPACE



Chebyshev Polynomials: $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$

METHOD

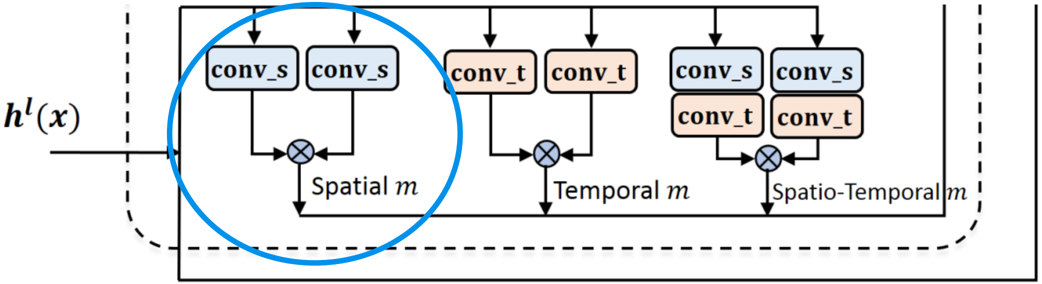


GCN SEARCH SPACE

The Chebyshev polynomial functions let the network determine which order and polynomial components each layer prefers

Chebyshev Polynomials: $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$

METHOD

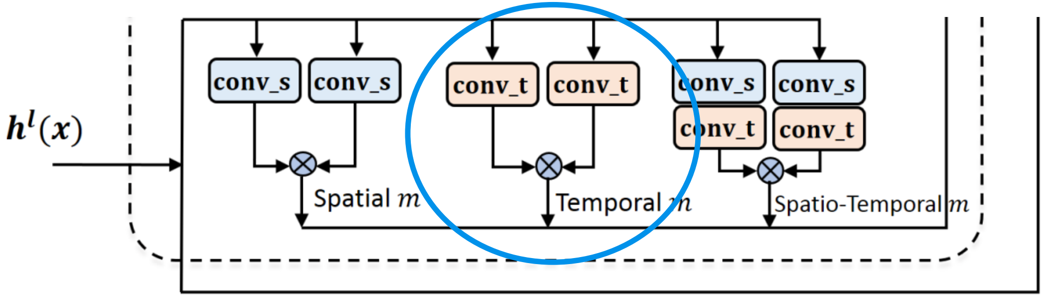


Structure Representation Correlation as **Spatial m**:

Compute based on the spatial node connections

Guassian Function: $$\forall i, j \in \mathcal{V}, A_D(i,j) = \frac{e^{\phi(h(x_i)) \bigotimes \psi(h(x_j))}}{\sum_{j=1}^{n} e^{\phi(h(x_i)) \bigotimes \psi(h(x_j))}}$$

METHOD



GCN

SEARCH

SPACE

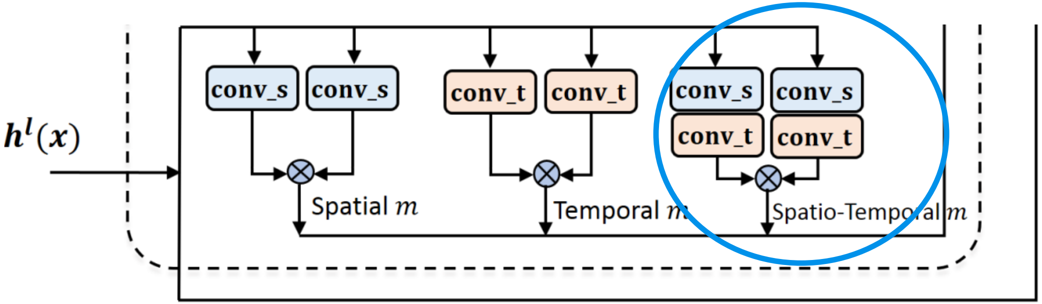
Temporal Representation Correlation as **Temporal m**:

Extract the temporal information of each node before computing node correlations.

Guassian Function:

$$\forall i,j \in \mathcal{V}, A_D(i,j) = \frac{e^{\phi(h(x_i)) \otimes \psi(h(x_j))}}{\sum_{j=1}^{n} e^{\phi(h(x_i)) \otimes \psi(h(x_j))}}$$

METHOD



Spatial-Temporal function: combining the Spatial and Temporal Module

Guassian Function:

$$\forall i, j \in \mathcal{V}, A_D(i,j) = \frac{e^{\phi(h(x_i)) \otimes \psi(h(x_j))}}{\sum_{j=1}^{n} e^{\phi(h(x_i)) \otimes \psi(h(x_j))}}$$

SEARCH STRATEGY

Cross-Entropy

CEIM

Importance-Mixing

CEIM

Three steps:

- **Sampling populations:** Modeling the architecture distribution with a Gaussian distribution.
- **Selecting populations:** Combining S.new with historical selected populations S.old
- **Updating:** Using new selected samples to update the architecture distribution.

(S.new: the populations for CEIM)

3.

EVALUATION

Let’s end with the proposed method evaluation

EVALUATION

Methods	Joint (%)	Bone (%)	Combine (%)
2S-AGCN	93.7	93.2	95.1
T	93.8	93.7	95.1
ST	94.0	93.8	95.2
T+Cheb	94.0	93.9	95.3
ST+Cheb	94.2	93.9	95.3
S+T+ST+Cheb	93.9	93.6	95.1
NAS	**94.6**	**94.7**	**95.7**

Performance Comparison on NTU RGB+D with CV evaluation

EVALUATION

Methods	CS (%)	CV (%)
DPRL	83.5	89.8
SR-TSL	84.8	92.4
STGR-GCN	86.9	92.3
GR-GCN	87.5	94.3
AS-GCN	86.8	94.2
2S-AGCN	88.5	95.1
NAS (Joint+Bone)	**89.4**	**95.7**

Performance Comparison on NTU RGB+D with six current state-of-the-art methods

EVALUATION

Methods	Top-1(%)	Top-5(%)
P-LSTM	16.4	25.8
ST-GCN	30.7	52.8
AS-GCN	34.8	56.5
2S-AGCN (Joint)	35.1	57.1
2S-AGCN (Bone)	33.3	55.7
2S-AGCN	36.1	58.7
NAS (Joint)	**35.5**	**57.9**
NAS (Bone)	**34.9**	**57.1**
NAS (Joint+Bone)	**37.1**	**60.1**

Performance Comparison on Kinetics with eight current state-of-the-art methods

SUMMARY

- Determine the graph convolution architecture with NAS for skeleton-based action recognition.
- Provide multiple dynamic graph substructures on the basis of various spatial-temporal graph modules
- Build higher-order connections with Chebyshev polynomial.
- Use CEIM for improving the search efficiency

THANKS FOR LISTENING!