Group Meeting - October 09, 2020

Paper review & Research progress

Truong Son Hy *

*Department of Computer Science
The University of Chicago

Ryerson Physical Lab

Bruce Lee

Empty your mind. Be formless, shapeless, like water. If you put water into a cup, it becomes the cup. You put water into a bottle, it becomes the bottle. You put it into a teapot, it becomes the teapot. Water can flow or it can crash. Be water.

Starved Rock, Summer 2020

Papers

1. **Unsupervised Joint k-node Graph Representations with Compositional Energy-Based Models** (NeurIPS 2020)
   `https://arxiv.org/abs/2010.04259`
2. **PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation** (CVPR 2017)
   `http://stanford.edu/~rqi/pointnet/`

Paper 1

Unsupervised Joint k-node Graph Representations with Compositional Energy-Based Models (NeurIPS 2020)

Leonardo Cotta, Carlos H. C. Teixeira, Ananthram Swami, Bruno Ribeiro

`https://arxiv.org/abs/2010.04259`

Proposals

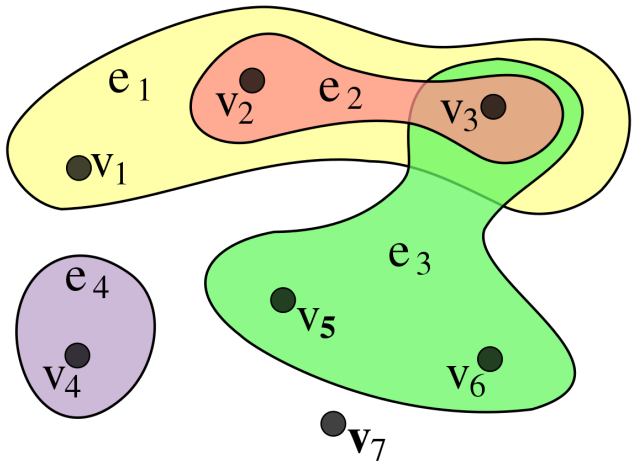
Proposals

MHM-GNN:

- MHM = Motif Hypergraph Markov
- An inductive unsupervised graph representation approach that combines:
  - Joint k-**node representations** with **energy-based models** (hypergraph Markov networks)
  - GNNs
- Loss is intractable $\rightarrow$ Optimize with an upper bound loss using a finite-sample unbiased Markov Chain Monte Carlo estimator.

Motivation

- For downstream tasks that require jointly reasoning about $k > 2$ nodes, but whose input data are dyadic relations (i.e., standard graphs).
- **For example:** Predict which k products could be jointly purchased in the same shopping cart. But the database only records *(product, product)* dyads to safeguard user information.
- Inductive unsupervised learning using GNNs in (dyadic) graphs is **restricted** to node and edge representations due to the edge-based losses. Graph AEs assume conditional independence of edges and can be classified as edge-based models.



Source: Wikipedia

Motif Hypergraph Markov (1)

Induced subgraphs are also referred to as **motifs**, **graphlets**, **graph fragments**, or **subgraphs**.

Induced subgraph

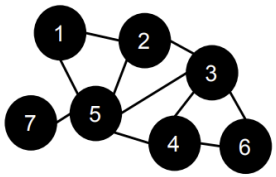
Let $C \subseteq V$: $|C| = k$ be a set of k nodes from V with corresponding sorted sequence

$$\vec{C} = [C_1, .., C_k] : C_i < C_{i+1} \quad \forall i \in \{1, .., k\}$$

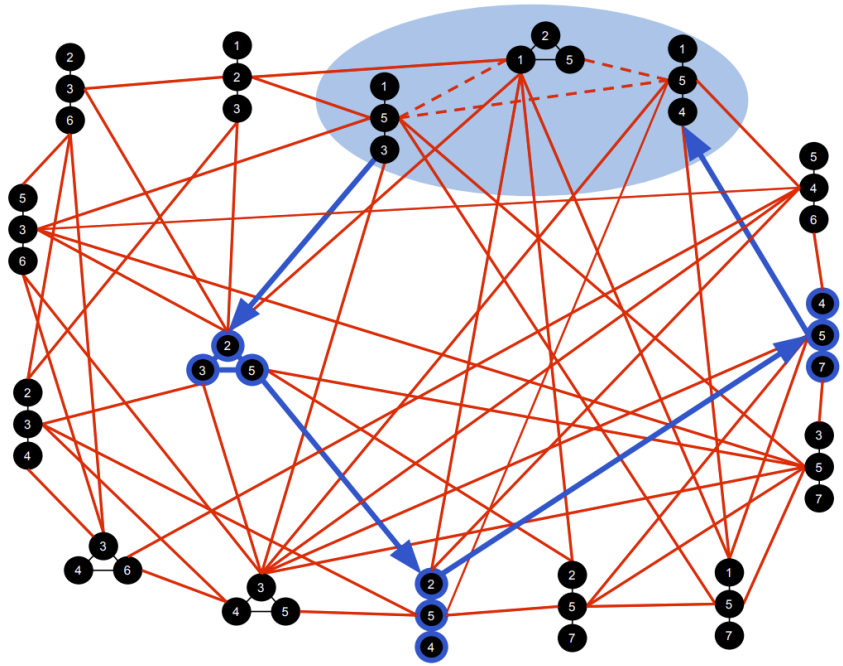
Then $G^{(C)} = (V^{(C)}, E^{(C)}, \boldsymbol{X}^{(C)})$ is the induced subgraph of C in G with:

- Set of vertices $V^{(C)} = \{1, .., k\}$.
- Adjacency matrix $\boldsymbol{A}^{(C)} \in \{0, 1\}^{k \times k}$: $\boldsymbol{A}^{(C)}_{ij} = \boldsymbol{A}_{C_i C_j}$.
- Node features $\boldsymbol{X}^{(C)} \in \mathbb{R}^{k \times p}$: $\boldsymbol{X}^{(C)}_{i,:} = \boldsymbol{X}_{C_i,:}$.

Motif Hypergraph Markov (2)



(a) Original graph.



(b) k-CNHON with a supernode of size 3 highlighted in blue and an RWT example. Dashed red edges exist in the k-HON but are removed in the k-CNHON.

Figure 2: A graph and its corresponding k-CNHON with an RWT example.

k-HON = higher-order network

k-CNHON = collapsed node HON

RWT = Random Walk Tour

Motif Hypergraph Markov (3)

Hypergraph Markov Networks (HMNs)

A hypergraph Markov network is a Markov network where the joint probability distribution of $\boldsymbol{Y} = \{Y_1, .., Y_\ell\}$ can be expressed as:

$$\mathbb{P}(\boldsymbol{Y} = \boldsymbol{y}) = \frac{1}{Z} \prod_{h \in \mathcal{H}} \phi_h(\boldsymbol{y}_h)$$

- The partition function

$$Z = \sum_{\boldsymbol{y}' \in \boldsymbol{Y}} \prod_{h \in \mathcal{H}} \phi_h(\boldsymbol{y}'_h)$$

- $\mathcal{H} \subseteq \mathcal{P}(\boldsymbol{Y}) - \{\emptyset\}$ is the set of hyper-edges in the Markov network, $\mathcal{P}(\boldsymbol{Y})$ is the powerset of a set $\boldsymbol{Y}$.
- $\boldsymbol{Y}_h$ are the random variables associated with hyper-edge h. $\boldsymbol{y}$, $\boldsymbol{y}_h$ are the assignments of $\boldsymbol{Y}$ and $\boldsymbol{Y}_h$, respectively.

Motif Hypergraph Markov (4)

Energy-based HMN assuming strictly positive potentials:

$$\mathbb{P}(\boldsymbol{Y} = \boldsymbol{y}) = \frac{1}{Z} \prod_{h \in \mathcal{H}} \exp(-\phi_h(\boldsymbol{y}_h)) = \frac{1}{Z} \exp\left(- \sum_{h \in \mathcal{H}} \phi_h(\boldsymbol{y}_h) \right)$$

MHM-GNN

Let $\mathcal{C}^{(k)}$ denote the set of all $\binom{n}{k}$ combinations of k nodes from G.

$$\mathbb{P}(\boldsymbol{A}, \boldsymbol{X} | \boldsymbol{W}) = \frac{\exp\left(- \sum_{C \in \mathcal{C}^{(k)}} \phi(\boldsymbol{A}^{(C)}, \boldsymbol{X}^{(C)}; \boldsymbol{W}) \right)}{Z(\boldsymbol{W})}$$

where $\boldsymbol{W}$ is the parameters and the partition function:

$$Z(\boldsymbol{W}) = \sum_{n=1}^{\infty} \sum_{\boldsymbol{A}' \in \{0,1\}^{n \times n}} \int_{\boldsymbol{X}' \in \mathbb{R}^{n \times p}} \exp\left(- \sum_{C \in \mathcal{C}^{(k)}} \phi(\boldsymbol{A}'^{(C)}, \boldsymbol{X}'^{(C)}; \boldsymbol{W}) \right) d\boldsymbol{X}'$$

Motif Hypergraph Markov (5)

Important reference about Exchangeability [44]

Bayesian Models of Graphs, Arrays and Other Exchangeable Random Structures (IEEE PAMI)

Peter Orbanz, Daniel M. Roy

`https://arxiv.org/abs/1312.7857`

MHM-GNN will learn a jointly exchangeable distribution [44] if the subgraph energy function $\phi(\cdot, \cdot; \boldsymbol{W})$ is jointly exchangable, such as a GNN.
$\Rightarrow \phi(\boldsymbol{A}^{(C)}, \boldsymbol{X}^{(C)}; \boldsymbol{W})$ needs to be jointly exchangeable with respect to the subgraph $G^{(C)}$.

Exchangeable random structures (1)

Definition – Exchangeable sequence

We call (X_i) exchangeable if its joint distribution satisfies:

$$\mathbb{P}(X_1 \in A_1, X_2 \in A_2, ..) = \mathbb{P}(X_{\pi(1)} \in A_1, X_{\pi(2)} \in A_2, ..)$$

for every permutation π of $\mathbb{N}$. We write:

$$(X_1, X_2, ..) =^d (X_{\pi(1)}, X_{\pi(2)}, ..)$$

Exchangeable random structures (2)

Theorem II.1 (de Finetti)**.** *Let* $(X_1, X_2, \dots)$ *be an infinite sequence of random variables with values in a space* $\mathbf{X}$*. The sequence* $X_1, X_2, \dots$ *is exchangeable if and only if there is a random probability measure* Θ *on* $\mathbf{X}$ *such that the* X_i *are conditionally i.i.d. given* Θ *and*

$$\mathbb{P}(X_1 \in A_1, X_2 \in A_2, \dots) = \int_{\mathbf{M}(\mathbf{X})} \prod_{i=1}^{\infty} \theta(A_i)\, \nu(\mathrm{d}\theta) \quad \text{(II.3)}$$

where ν *is the distribution of* Θ*.* □

The integral on the right-hand side of (II.3) can be interpreted as a two-stage sampling procedure:

1) Sample $\Theta \sim \nu$, i.e., draw a probability distribution at random from the distribution ν.
2) Conditioned on Θ, sample the X_n conditionally i.i.d. as

$$X_1, X_2, \dots | \Theta \sim_{\text{iid}} \Theta\ . \quad \text{(II.4)}$$

Exchangeable random structures (3)

Exchangeable 2d-array

A random 2d-array (X_{ij}) is called **jointly exchangeable** if:

$$(X_{ij}) =^d (X_{\pi(i)\pi(j)})$$

for every permutation π, and **separately exchangeable** if:

$$(X_{ij}) =^d (X_{\pi(i)\pi'(j)})$$

for every pair of permutations π and π'.

Exchangeable random structures (4)

Theorem III.2 (Aldous-Hoover). *A random array* (X_{ij}) *is jointly exchangeable if and only if it can be represented as follows: There is a random function* $F : [0,1]^3 \to \mathbf{X}$ *such that*

$$(X_{ij}) \stackrel{d}{=} (F(U_i, U_j, U_{\{i,j\}})) \,, \tag{III.4}$$

where $(U_i)_{i\in\mathbb{N}}$ *and* $(U_{\{i,j\}})_{i,j\in\mathbb{N}}$ *are, respectively, a sequence and an array of i.i.d.* Uniform$[0,1]$ *random variables, which are independent of* F. □

Exchangeable random structures (5)

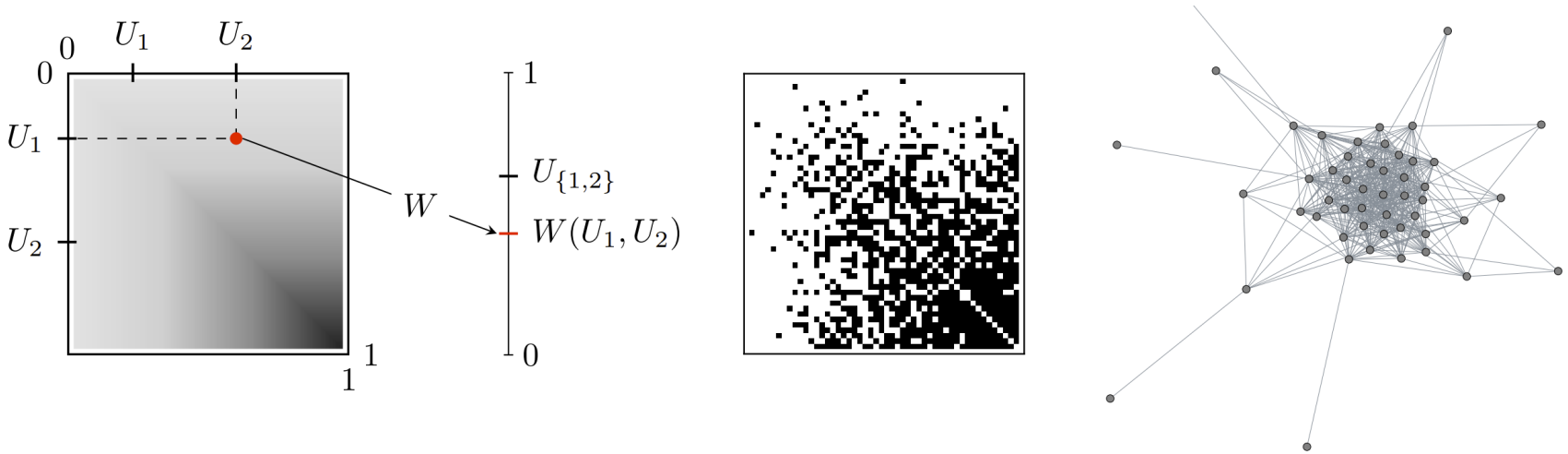


Fig. 4: Sampling an exchangeable random graph according to Eq. (III.8). *Left:* A heat-map visualization of the random function W on $[0,1]^2$, given here by $W(x,y) = \min\{x,y\}$. Darker shades represent larger values. In the case depicted here, the edge $\{1,2\}$ is not present in the graph, because $U_{\{1,2\}} > W(U_1, U_2)$. *Middle:* The adjacency matrix of a 50-vertex random graph, sampled from the function on the left. Rows (and columns) in the matrix have been ordered by their underlying U_i value, resulting in a matrix resembling W. *Right:* A plot of the random graph sample. The highly connected vertices plotted in the center correspond to values lower right region in $[0,1]^2$. The minimum function example, due to Lovász [47], is chosen as a particularly simple symmetric function which is not piece-wise constant. See Section IV for examples with more structure.

Exchangeable random structures (6)

From the **Aldous-Hoover theorem**:

Corollary III.6. *Let G be a random simple graph with vertex set $\mathbb{N}$ and let X be its adjacency matrix. Then G is an exchangeable graph if and only if there is a random function W from $[0,1]^2$ to $[0,1]$ such that*

$$(X_{ij}) \stackrel{d}{=} (\mathbb{1}\{U_{\{i,j\}} < W(U_i, U_j)\}), \qquad \text{(III.8)}$$

where U_i and $U_{\{i,j\}}$ are independent i.i.d. uniform variables as in (III.4), which are independent of W. □

The representation (III.8) yields the following generative process:

1) Sample a random function $W \sim \nu$.
2) For every vertex $i \in \mathbb{N}$, sample an independent uniform random variable U_i, independent also from W.
3) For every pair of vertices $i < j \in \mathbb{N}$, sample

$$X_{ij} \mid W, U_i, U_j \sim \text{Bernoulli}(W(U_i, U_j)) \,, \qquad \text{(III.9)}$$

where $X_{ij} = 1$ indicates the edge connecting i and j is present; if $X_{ij} = 0$, it is absent.

We call the symmetric random function W from $[0,1]^2$ to $[0,1]$ as **graphon**. Statistical models of exchangeable simple graphs are parameterized by graphons.

Motif Hypergraph Markov (6)

Permutation invariant subgraph representation:

$$h^{(C)}(\boldsymbol{A}^{(C)}, \boldsymbol{X}^{(C)}; \boldsymbol{W}_{\text{GNN}}, \boldsymbol{W}_R) = \text{READOUT}(\text{GNN}(\boldsymbol{A}^{(C)}, \boldsymbol{X}^{(C)}; \boldsymbol{W}_{\text{GNN}}); \boldsymbol{W}_R)$$

READOUT = function over the nodes' representations (given by a GNN). For example, a row-wise sum followed by a MLP.

The energy of a subgraph $G^{(C)}$ is defined as:

$$\phi(\boldsymbol{A}^{(C)}, \boldsymbol{X}^{(C)}; \boldsymbol{W}) = \boldsymbol{W}_{\text{energy}}^T \rho(h^{(C)}(\boldsymbol{A}^{(C)}, \boldsymbol{X}^{(C)}; \boldsymbol{W}_{\text{GNN}}, \boldsymbol{W}_R); \boldsymbol{W}_\rho)$$

where $\rho(\cdot; \boldsymbol{W}_\rho)$ is a permutation sensitive function (?) with parameters $\boldsymbol{W}_\rho$ such as a multi-layer perceptron with range in $\mathbb{R}^{1\times H}$ and $\boldsymbol{W}_{\text{energy}} \in \mathbb{R}^{1\times H}$.

Learning MHM-GNNs (1)

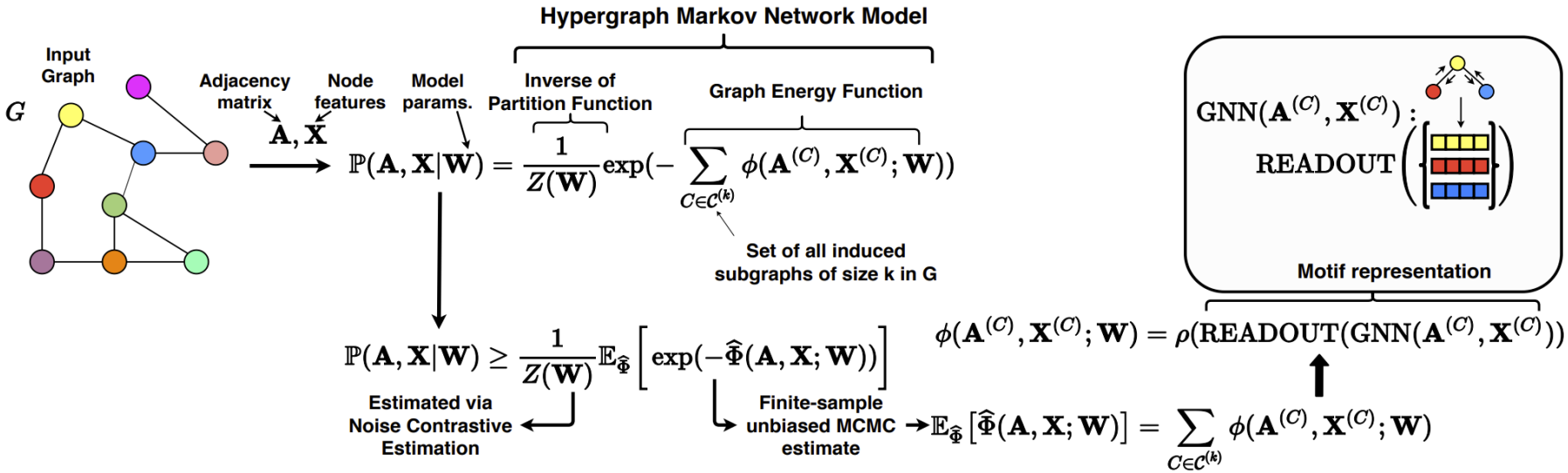


Figure 1: The proposed unsupervised graph representation using motif compositions. Here, we present the MHM-GNN model from Equation (1), the energy estimator $\widehat{\Phi}$ from Equation (4), the motif energy and representation from Equation (2).

Learning MHM-GNNs (2)

Directly computing $Z(\boldsymbol{W})$ is **intractable**.
$\Rightarrow$ Noise-Contrastive Estimation (NCE). Motivated by GAN?

$$\mathcal{L}(\boldsymbol{A}, \boldsymbol{X}; \boldsymbol{W}) = -\sum_{\boldsymbol{A} \in \mathcal{D}_{\text{true}}} \log(\hat{y}(\Phi(\boldsymbol{A}, \boldsymbol{X}; \boldsymbol{W}), \mathbb{P}_n(\boldsymbol{A}, \boldsymbol{X})))$$

$$-\sum_{\boldsymbol{A} \in \mathcal{D}_{\text{true}}} \log(1 - \hat{y}(\Phi(\boldsymbol{A}, \boldsymbol{X}; \boldsymbol{W}), \mathbb{P}_n(\boldsymbol{A}, \boldsymbol{X})))$$

where:

$$\hat{y}(\Phi(\boldsymbol{A}, \boldsymbol{X}; \boldsymbol{W}), \mathbb{P}_n(\boldsymbol{A}, \boldsymbol{X})) = \sigma(-\Phi(\boldsymbol{A}, \boldsymbol{X}; \boldsymbol{W}) - \log(M\mathbb{P}_n(\boldsymbol{A}, \boldsymbol{X})))$$

and

$$\Phi(\boldsymbol{A}, \boldsymbol{X}; \boldsymbol{W}) = \sum_{C \in \mathcal{C}^{(k)}} \phi(\boldsymbol{A}^{(C)}, \boldsymbol{X}^{(C)}; \boldsymbol{W})$$

denotes the **total energy**.

Learning MHM-GNNs (3)

Estimate the total energy:

$$\Phi(\boldsymbol{A}, \boldsymbol{X}; \boldsymbol{W}) = \sum_{C \in \mathcal{C}_{\text{conn}}^{(k)}} \phi(\boldsymbol{A}^{(C)}, \boldsymbol{X}^{(C)}; \boldsymbol{W}) + \text{constant}$$

on **connected induced subgraphs** (CIS), assuming some constant energy for disconnected subgraphs.

Learning MHM-GNNs (4)

Theorem 1. *Let $G^{(k)}$ be the k-HON of a graph G, a set $\mathcal{I}$ of k-node sets that induce CISes in G (as described above) and $N^{(k)}(C)$ the set of neighbors of the corresponding node of CIS C in $G^{(k)}$. In addition, consider the sample-path $\mathcal{T}^r = (v_1^r, ..., v_{t^r}^r, v_{t^r+1}^r)$ visited by the r-th RWT on $G^{(k,\mathcal{I})}$ starting from supernode $v_{\mathcal{I}}^{(k)}$, where v_i^r is the node reached at step i for $1 \leq r \leq q$ (Definition 4), and $q \geq 1$ is the number of RWTs. Since $\mathcal{T}^r$ is a RWT, $v_1^r = v_{\mathcal{I}}^{(k)}$, $v_{t^r+1}^r = v_{\mathcal{I}}^{(k)}$ and $v_i^r \neq v_{\mathcal{I}}^{(k)} : 1 < i < t^r + 1$. The nodes $(v_2^r, ..., v_{t^r}^r)$ in the sample path $\mathcal{T}^r$ have a corresponding sequence of induced k-node subgraphs in the graph G, denoted $\mathcal{T}_C^r = (C_i^r)_{i=2}^{t^r}$. Then, the estimator*

$$\widehat{\Phi}(\mathbf{A}, \boldsymbol{X}; \mathbf{W}) = \underbrace{\sum_{v \in \mathcal{I}} \phi(\mathbf{A}^{(v)}, \boldsymbol{X}^{(v)}; \mathbf{W})}_{\text{Energy of } k\text{-node CISes in } \mathcal{I} \text{ (supernode)}} + \underbrace{\left(\frac{\sum_{u \in \mathcal{I}} |N^{(k)}(u) \backslash \mathcal{I}|}{q}\right) \sum_{r=1}^{q} \sum_{i=2}^{t^r} \frac{\phi(\mathbf{A}^{(C_i^r)}, \boldsymbol{X}^{(C_i^r)}; \mathbf{W})}{|N^{(k)}(C_i^r)|}}_{\text{RWT-estimated energy of remaining } k\text{-node CISes in } G}$$

is an unbiased and consistent estimator of $\Phi(\mathbf{A}, \boldsymbol{X}; \mathbf{W})$ in Equation (3) with constant=0.

$$\mathbb{E}_{\hat{\Phi}}[\hat{\mathcal{L}}(\boldsymbol{A}, \boldsymbol{X}; \boldsymbol{W})] \geq \mathcal{L}(\boldsymbol{A}, \boldsymbol{X}; \boldsymbol{W})$$

Paper 2

PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation (CVPR 2017)

Charles R. Qi, Hao Su, Kaichun Mo, Leonidas J. Guibas

PointNet (1)

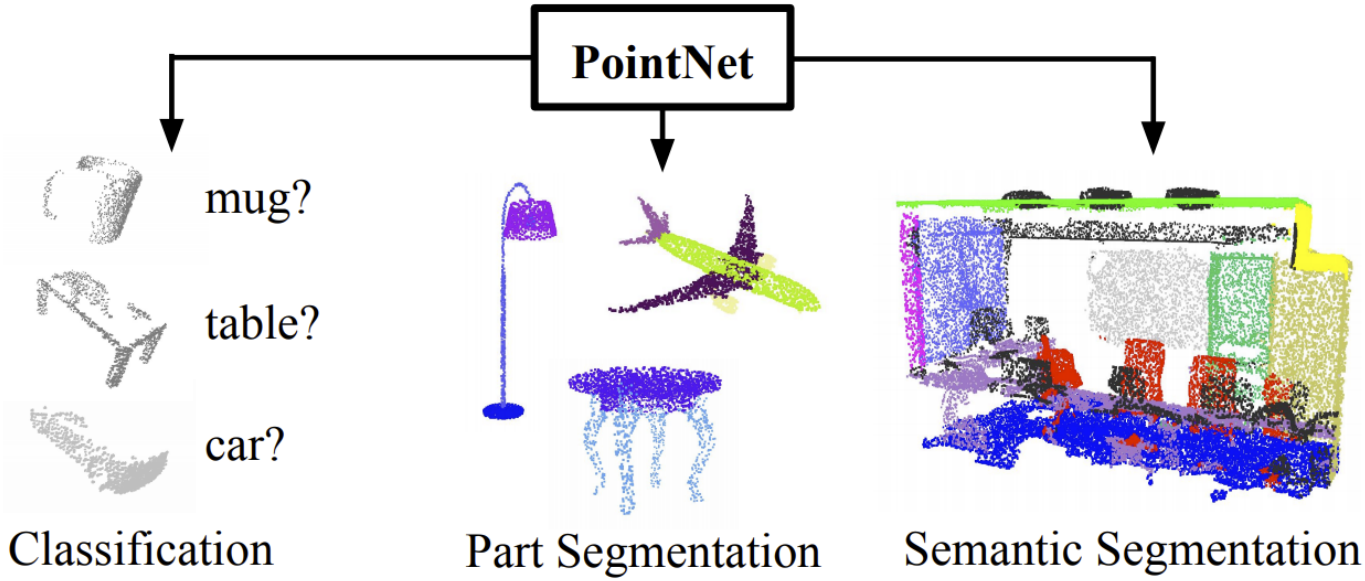


Figure 1. **Applications of PointNet.** We propose a novel deep net architecture that consumes raw point cloud (set of points) without voxelization or rendering. It is a unified architecture that learns both global and local point features, providing a simple, efficient and effective approach for a number of 3D recognition tasks.

PointNet (2)

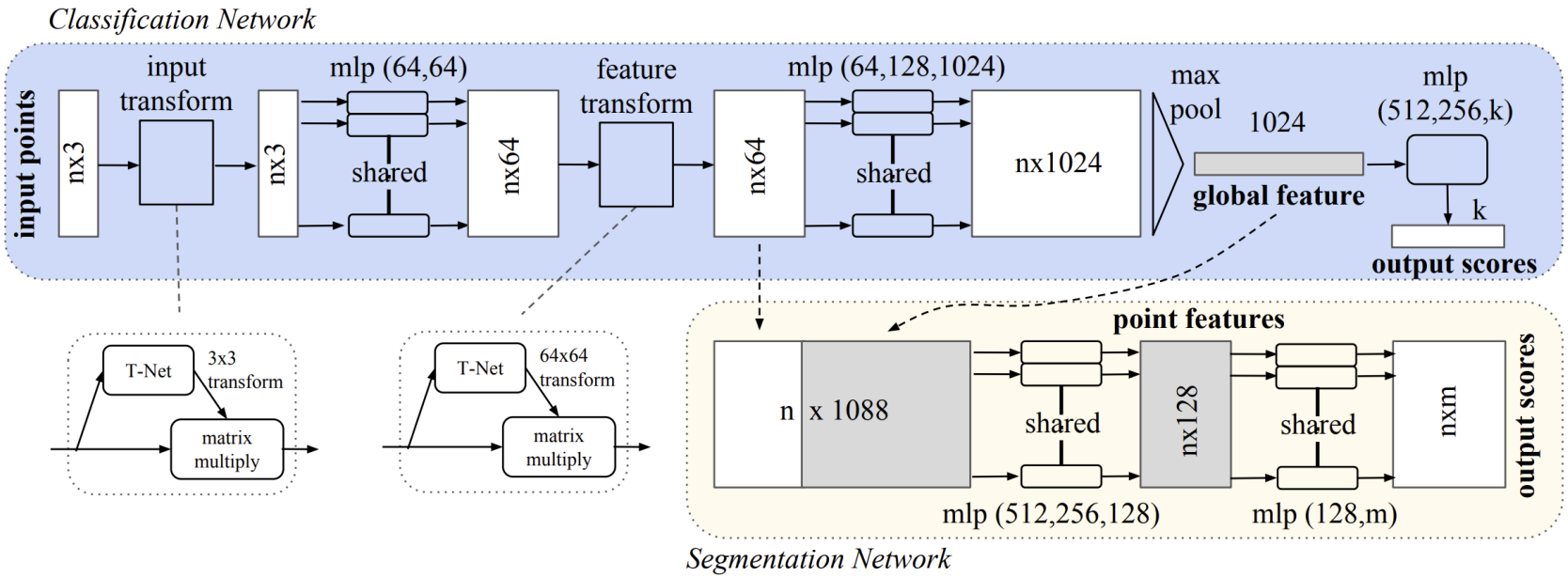


Figure 2. **PointNet Architecture.** The classification network takes n points as input, applies input and feature transformations, and then aggregates point features by max pooling. The output is classification scores for k classes. The segmentation network is an extension to the classification net. It concatenates global and local features and outputs per point scores. "mlp" stands for multi-layer perceptron, numbers in bracket are layer sizes. Batchnorm is used for all layers with ReLU. Dropout layers are used for the last mlp in classification net.

PointNet (3)

	input	#views	accuracy avg. class	accuracy overall
SPH [11]	mesh	-	68.2	-
3DShapeNets [28]	volume	1	77.3	84.7
VoxNet [17]	volume	12	83.0	85.9
Subvolume [18]	volume	20	86.0	**89.2**
LFD [28]	image	10	75.5	-
MVCNN [23]	image	80	**90.1**	-
Ours baseline	point	-	72.6	77.4
Ours PointNet	point	1	86.2	**89.2**

Table 1. **Classification results on ModelNet40.** Our net achieves state-of-the-art among deep nets on 3D input.

PointNet (4)

	mean	aero	bag	cap	car	chair	ear phone	guitar	knife	lamp	laptop	motor	mug	pistol	rocket	skate board	table
# shapes		2690	76	55	898	3758	69	787	392	1547	451	202	184	283	66	152	5271
Wu [27]	-	63.2	-	-	-	73.5	-	-	-	74.4	-	-	-	-	-	-	74.8
Yi [29]	81.4	81.0	78.4	77.7	**75.7**	87.6	61.9	**92.0**	85.4	**82.5**	**95.7**	**70.6**	91.9	**85.9**	53.1	69.8	75.3
3DCNN	79.4	75.1	72.8	73.3	70.0	87.2	63.5	88.4	79.6	74.4	93.9	58.7	91.8	76.4	51.2	65.3	77.1
Ours	**83.7**	**83.4**	**78.7**	**82.5**	74.9	**89.6**	**73.0**	91.5	**85.9**	80.8	95.3	65.2	**93.0**	81.2	**57.9**	**72.8**	**80.6**

Table 2. **Segmentation results on ShapeNet part dataset.** Metric is mIoU(%) on points. We compare with two traditional methods [27] and [29] and a 3D fully convolutional network baseline proposed by us. Our PointNet method achieved the state-of-the-art in mIoU.

Baselines:

- (Wu et al., 2014) Interactive shape cosegmentation via label propagation
- (Yi et al., 2016) A Scalable Active Framework for Region Annotation in 3D Shape Collections

PointNet (5)

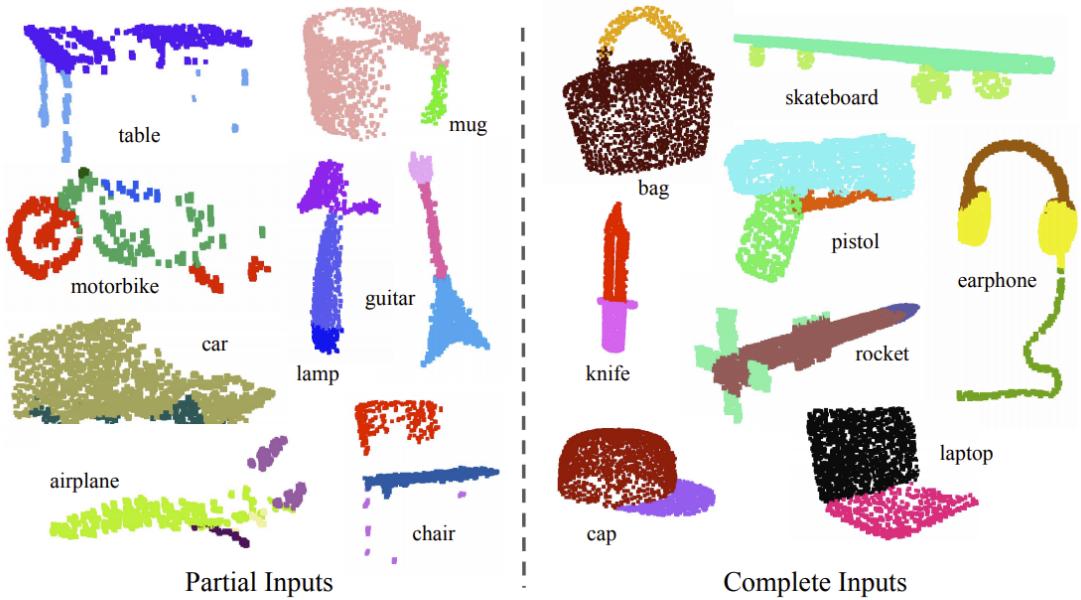


Figure 3. **Qualitative results for part segmentation.** We visualize the CAD part segmentation results across all 16 object categories. We show both results for partial simulated Kinect scans (left block) and complete ShapeNet CAD models (right block).

PointNet (6)

	mean IoU	overall accuracy
Ours baseline	20.12	53.19
Ours PointNet	**47.71**	**78.62**

Table 3. **Results on semantic segmentation in scenes.** Metric is average IoU over 13 classes (structural and furniture elements plus clutter) and classification accuracy calculated on points.

	table	chair	sofa	board	mean
# instance	455	1363	55	137	
Armeni et al. [1]	46.02	16.15	**6.78**	3.91	18.22
Ours	**46.67**	**33.80**	4.76	**11.72**	**24.24**

Table 4. **Results on 3D object detection in scenes.** Metric is average precision with threshold IoU 0.5 computed in 3D volumes.

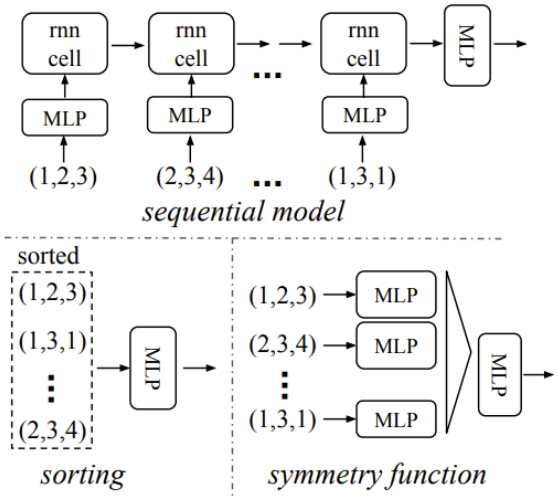
Baseline:

- (Armeni et al., 2016) 3d semantic parsing of large-scale indoor spaces

PointNet (7)

Figure 4. **Qualitative results for semantic segmentation.** Top row is input point cloud with color. Bottom row is output semantic segmentation result (on points) displayed in the same camera viewpoint as input.

PointNet (8)



	accuracy
MLP (unsorted input)	24.2
MLP (sorted input)	45.0
LSTM	78.5
Attention sum	83.0
Average pooling	83.8
Max pooling	**87.1**

Figure 5. **Three approaches to achieve order invariance.** Multi-layer perceptron (MLP) applied on points consists of 5 hidden layers with neuron sizes 64,64,64,128,1024, all points share a single copy of MLP. The MLP close to the output consists of two layers with sizes 512,256.

Note: This is where our research in permutation and rotational equivariance to fit in.