Geometric Deep Learning

Deep learning methods in Non-Euclidean domain

Jingru Li 06/17/2019

Outline

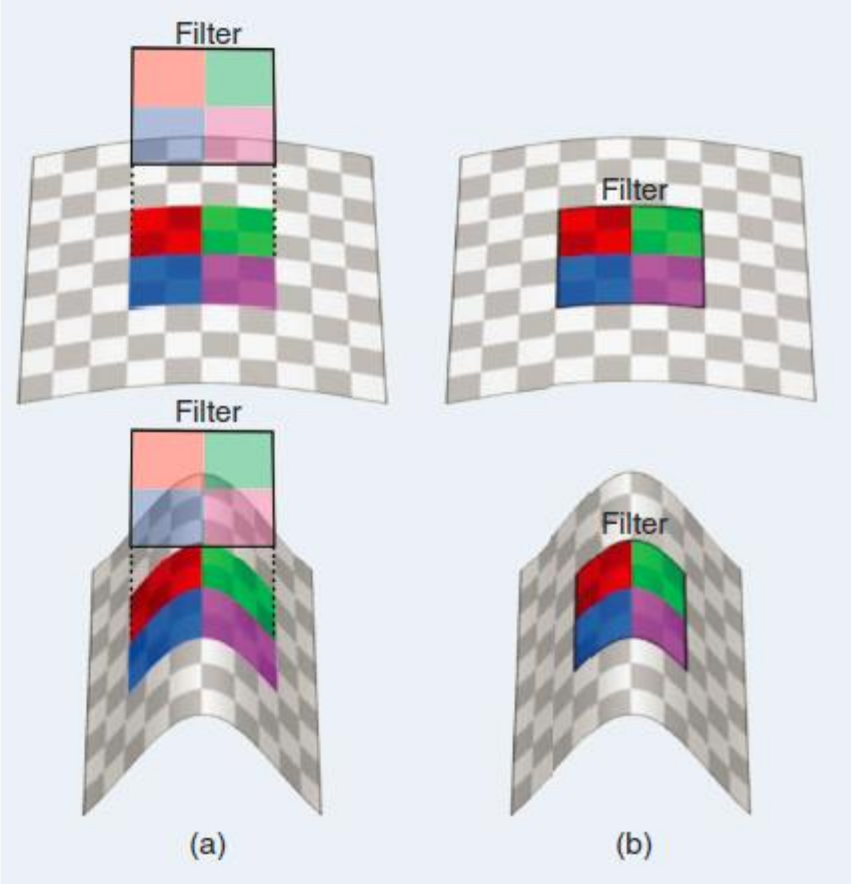
- Introduction

- Non-Euclidean domains
  - Manifolds
  - Graphs

- CNN + non-Euclidean domain
  - Spatial Methods
  - Spectral methods

- Applications
  - 3D modeling
  - Recommendation Systems
  - Network analysis

Motivation



Deep Learning on Euclidean

the invariances of these structures are built into networks used to model them

Deep Learning on non-Euclidean

Definition, targets ,Gradients, Integrations, etc.

Back to Convolution

A CNN layer consists of several convolutional layers of the form $\mathbf{g} = C_\Gamma(\mathbf{f})$, acting on a p-dimensional input $\mathbf{f}(x) = (f_1(x), \dots, f_p(x))$, and applying a bank of filters $\Gamma = (\gamma_{l,l'})$

$$g_l(x) = \xi\left(\sum_{l'=1}^{p} (f_{l'} \star \gamma_{l,l'})(x)\right)$$

Here the definition of convolution and pooling:

$$(f \star \gamma)(x) = \int_\Omega f(x - x')\gamma(x')dx'$$

$$g_l(x) = P(\{f_l(x') : x' \in N(x)\}), \quad l = 1, \dots, q.$$

Bronstein M M, Bruna J, LeCun Y, et al. Geometric deep learning: going beyond euclidean data[J]. IEEE Signal Processing Magazine, 2017, 34(4): 18-42.

Non-Euclidean domains

Manifold domain

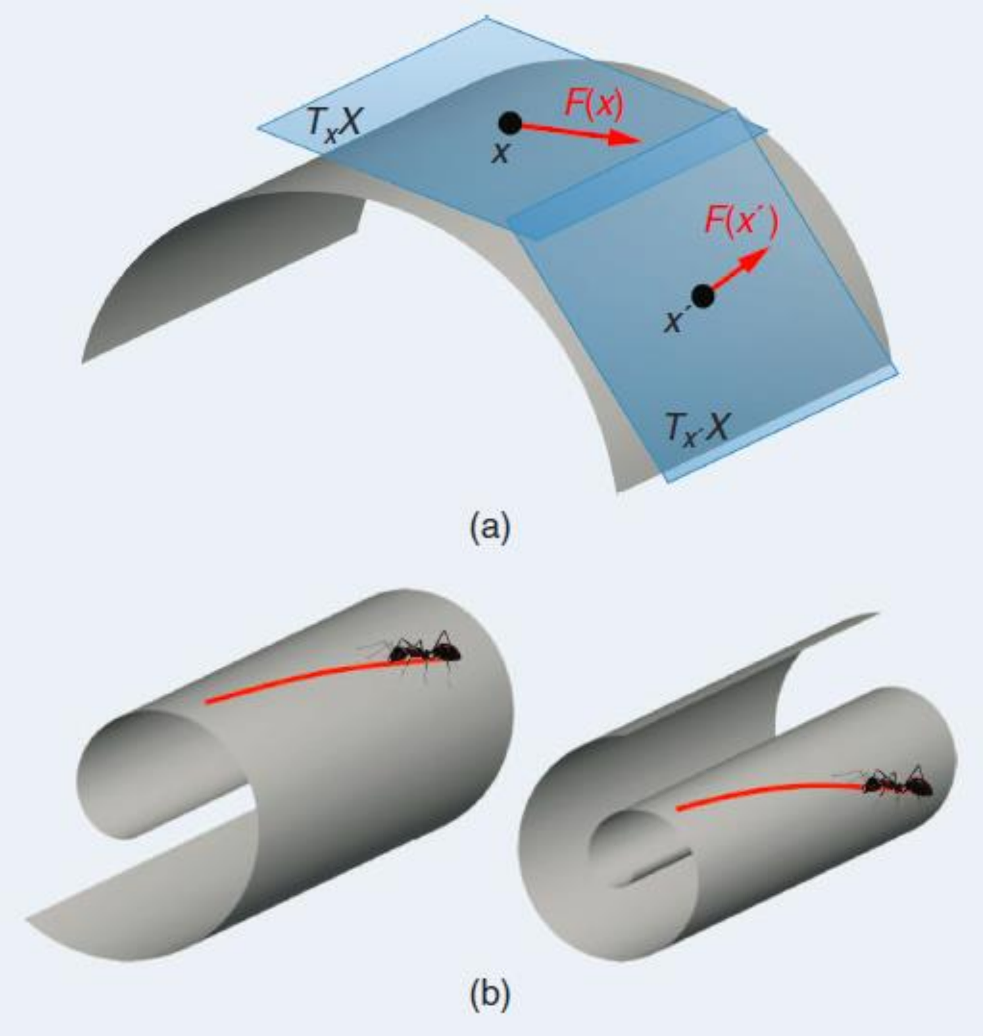
A topological space where each point x has a neighborhood that is topologically equivalent (homeomorphic) to a d-dimensional Euclidean space.

Calculus on manifolds

$$\langle f,g \rangle_{L^2(X)} = \int_X f(x)g(x)dx,$$

$$\langle F,G \rangle_{L^2(TX)} = \int_X \langle F(x),G(x)\rangle_{T_xX}dx.$$

$$\langle F,\nabla f \rangle_{L^2(TX)} = \langle \nabla^* F,f \rangle_{L^2(X)} = \langle -\mathrm{div}F,f \rangle_{L^2(X)}.$$



(a)
(b)

Bronstein M M, Bruna J, LeCun Y, et al. Geometric deep learning: going beyond euclidean data[J]. IEEE Signal Processing Magazine, 2017, 34(4): 18-42.

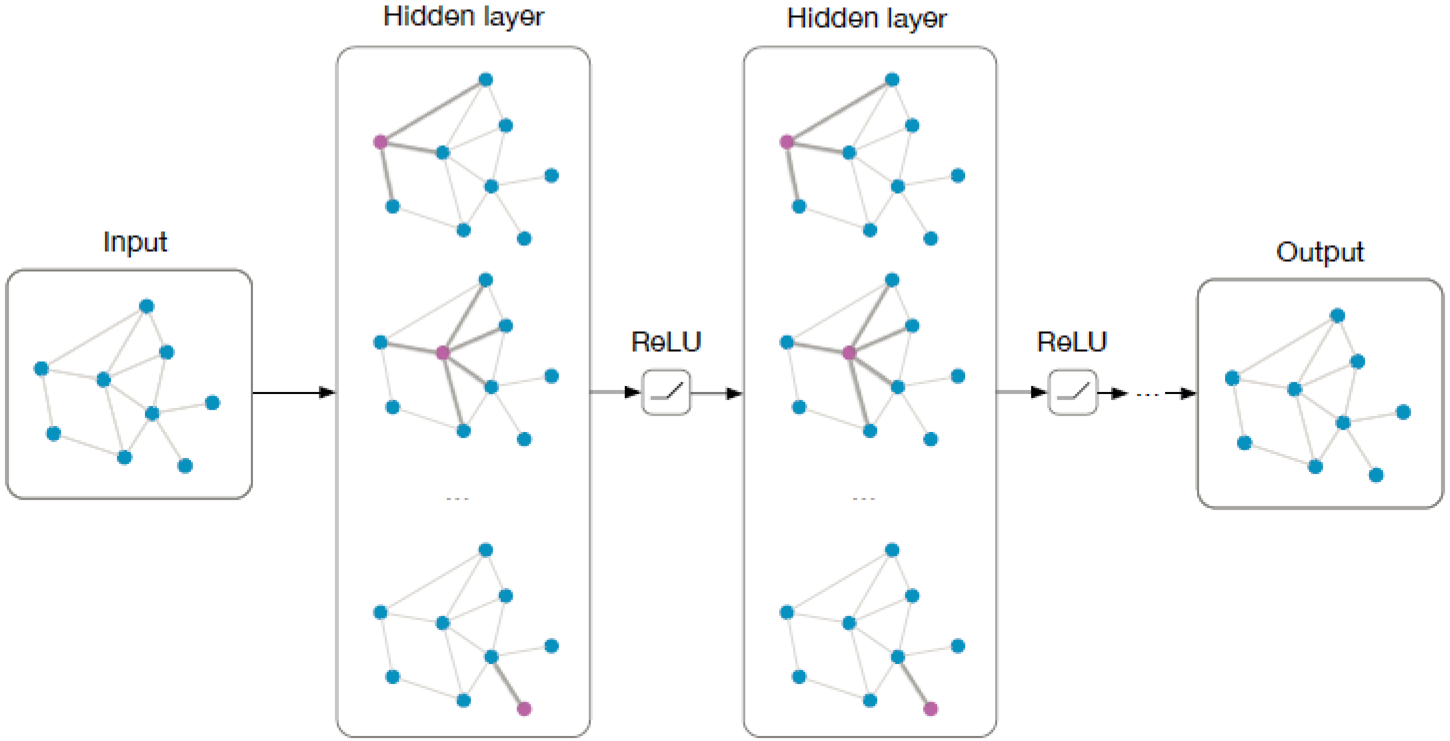
Non-Euclidean domains

Graph domain

Vert

Edg

Inne



Gra

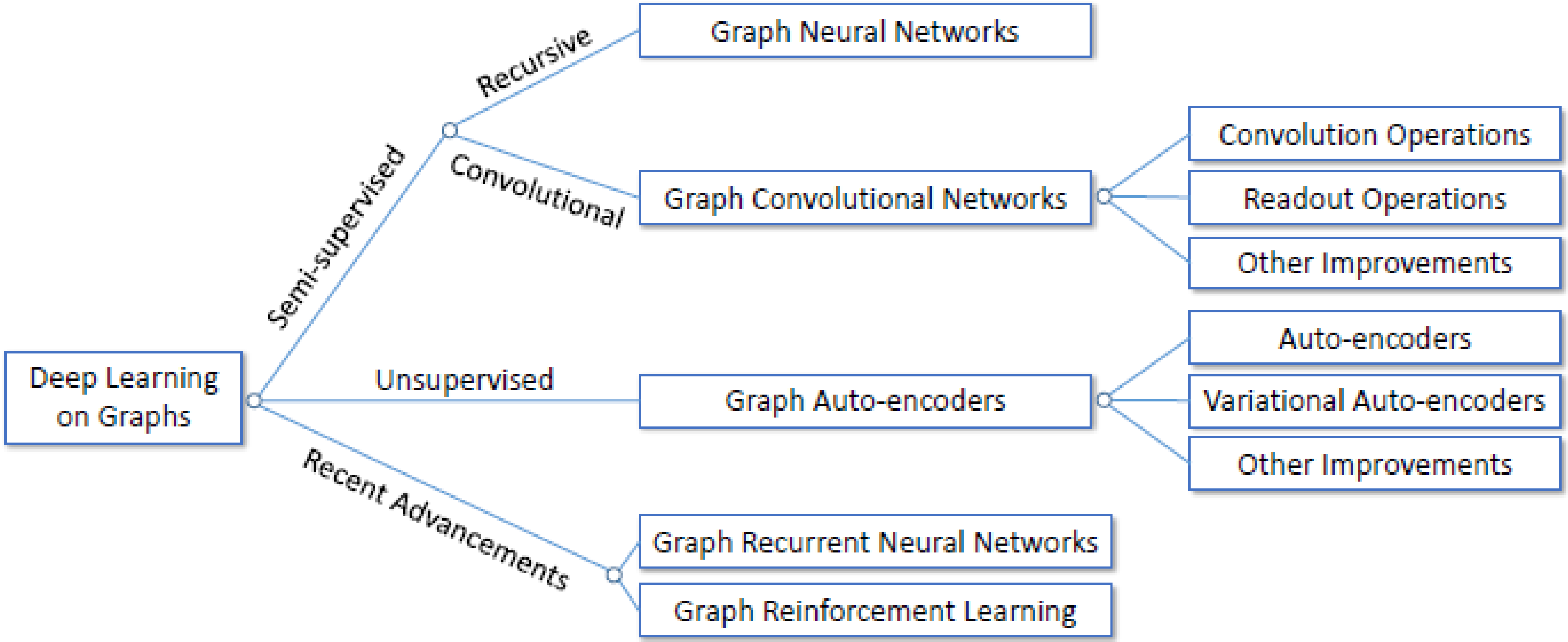
$$(\nabla f)_{ij} = f_i - f_j \qquad (\mathrm{div}F)_i = \frac{1}{a_i}\sum_{j:(i,j)\in\mathcal{E}} w_{ij}F_{ij} \qquad \Delta\mathbf{f} = \mathbf{A}^{-1}(\mathbf{D}-\mathbf{W})\mathbf{f}.$$

T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in Proceedings of the 5th International Conference on Learning Representations, 2017

Bronstein M M, Bruna J, LeCun Y, et al. Geometric deep learning: going beyond euclidean data[J]. IEEE Signal Processing Magazine, 2017, 34(4): 18-42

Non-Euclidean domains

Deep learning methods in graph



Fourier Transformation

Discrete Fourier Transformation:

$$f(x) = \sum_{i \geq 0} \underbrace{\langle f, \phi_i \rangle_{L^2(\mathcal{X})}}_{\hat{f}_i} \phi_i(x)$$

$$\Delta \phi_i = \lambda_i \phi_i, i = 0, 1, \ \ldots.$$

Here eigenvalue λ_i are real numbers, referred as spectrum.
According to convolutional theorem,

$$(\widehat{f \star g})(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx \int_{-\infty}^{\infty} g(x) e^{-i\omega x} dx.$$

$$(f \star g)(x) = \sum_{i \geq 0} \langle f, \phi_i \rangle_{L^2(\mathcal{X})} \langle g, \phi_i \rangle_{L^2(\mathcal{X})} \phi_i(x)$$

Bronstein M M, Bruna J, LeCun Y, et al. Geometric deep learning: going beyond euclidean data[J]. IEEE Signal Processing Magazine, 2017, 34(4): 18-42.

CNN + non-Euclidean

How to define convolution and pooling operation?

(1) Based on spectrum. (and GNN, GCN, GAT, etc.)

$$\mathbf{g}_l = \xi\left(\sum_{l'=1}^{q} \Phi_k \Gamma_{l,l'} \Phi_k^\top \mathbf{f}_{l'}\right)$$

$$\tilde{\Phi} \approx \mathbf{P}\Phi\begin{pmatrix}\mathbf{I}_{\alpha n} \\ 0\end{pmatrix}.$$

Drawbacks:

Laplacian based on Failed to adapt the model across different domains.

Bronstein M M, Bruna J, LeCun Y, et al. Geometric deep learning: going beyond euclidean data[J]. IEEE Signal Processing Magazine, 2017, 34(4): 18-42.

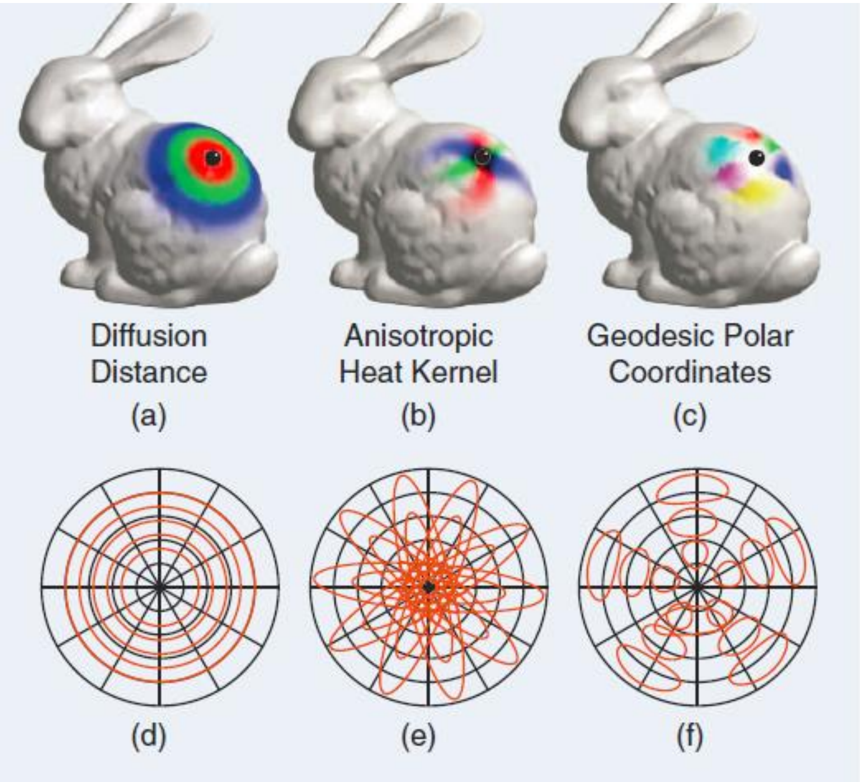
CNN + non-Euclidean

(2) Based on spatial.

$$D_j(x)f = \int_X f(x')v_j(x,x')dx', j = 1,\ldots,J,$$

$$(f \star g)(x) = \sum_j g_j D_j(x)f.$$

Examples: Geodesic CNN, Anisotropic CNN

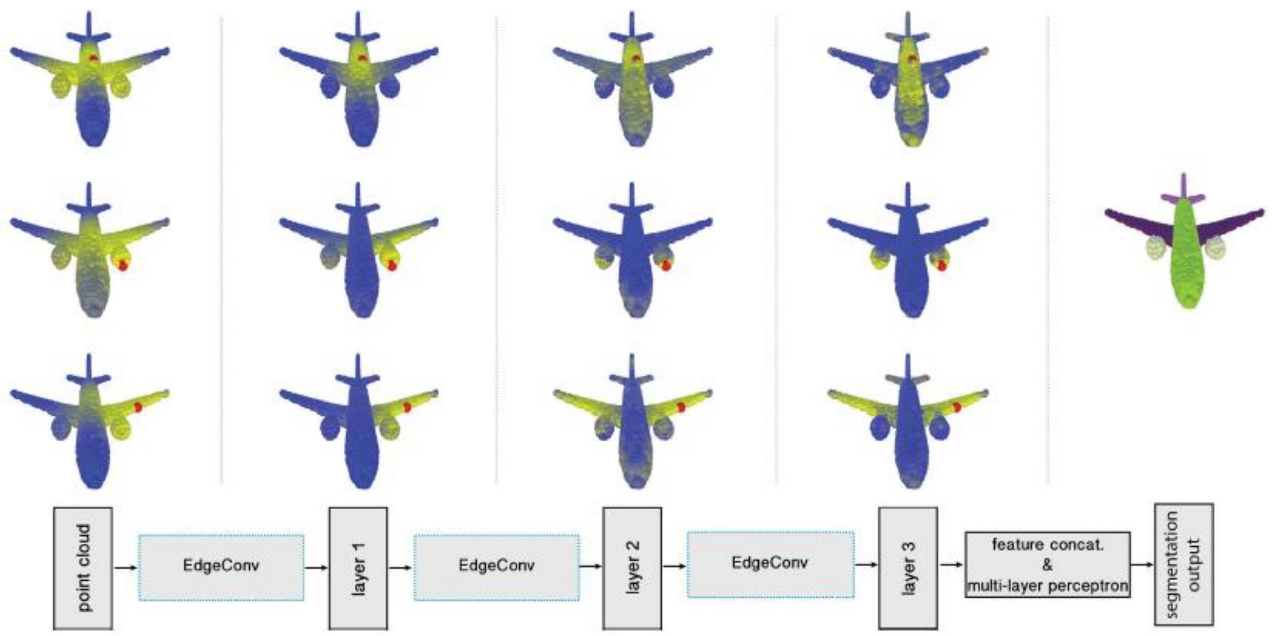


J. Masci, D. Boscaini, M. M. Bronstein, and P. Vandergheynst, “Geodesic convolutional neural networks on Riemannian manifolds,” in Proc. Int. IEEE Workshop 3-D Representation and Recognition (3DRR), 2015, pp. 832–840.

D. Boscaini, J. Masci, E. Rodolà, and M. M. Bronstein, “Learning shape correspondence with anisotropic convolutional neural networks,” in Proc. Int. Conf. Neural Information Processing Systems (NIPS), 2016, pp. 3189–3197

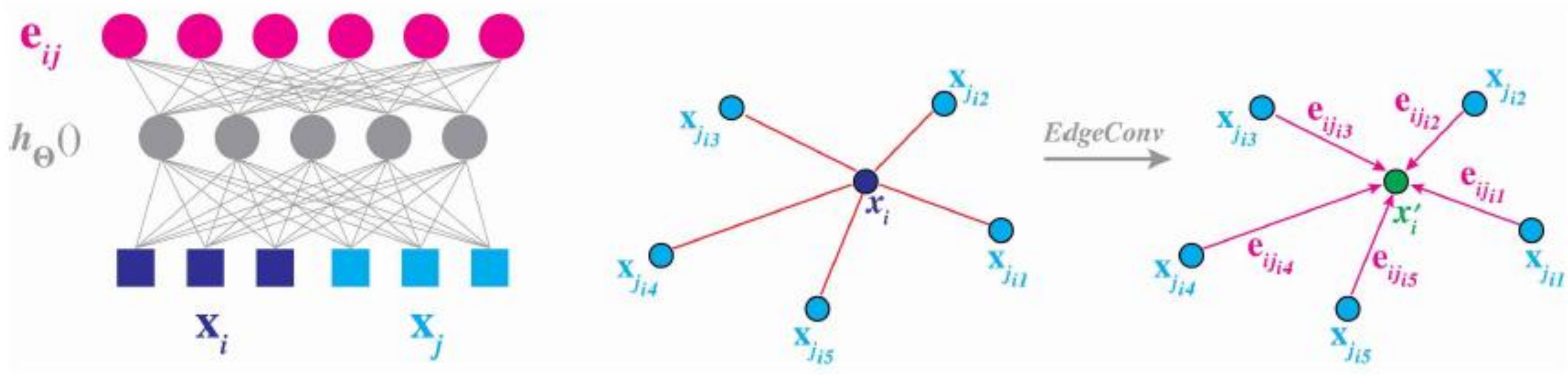
Bronstein M M, Bruna J, LeCun Y, et al. Geometric deep learning: going beyond euclidean data[J]. IEEE Signal Processing Magazine, 2017, 34(4): 18-42.

Application – 3D point cloud



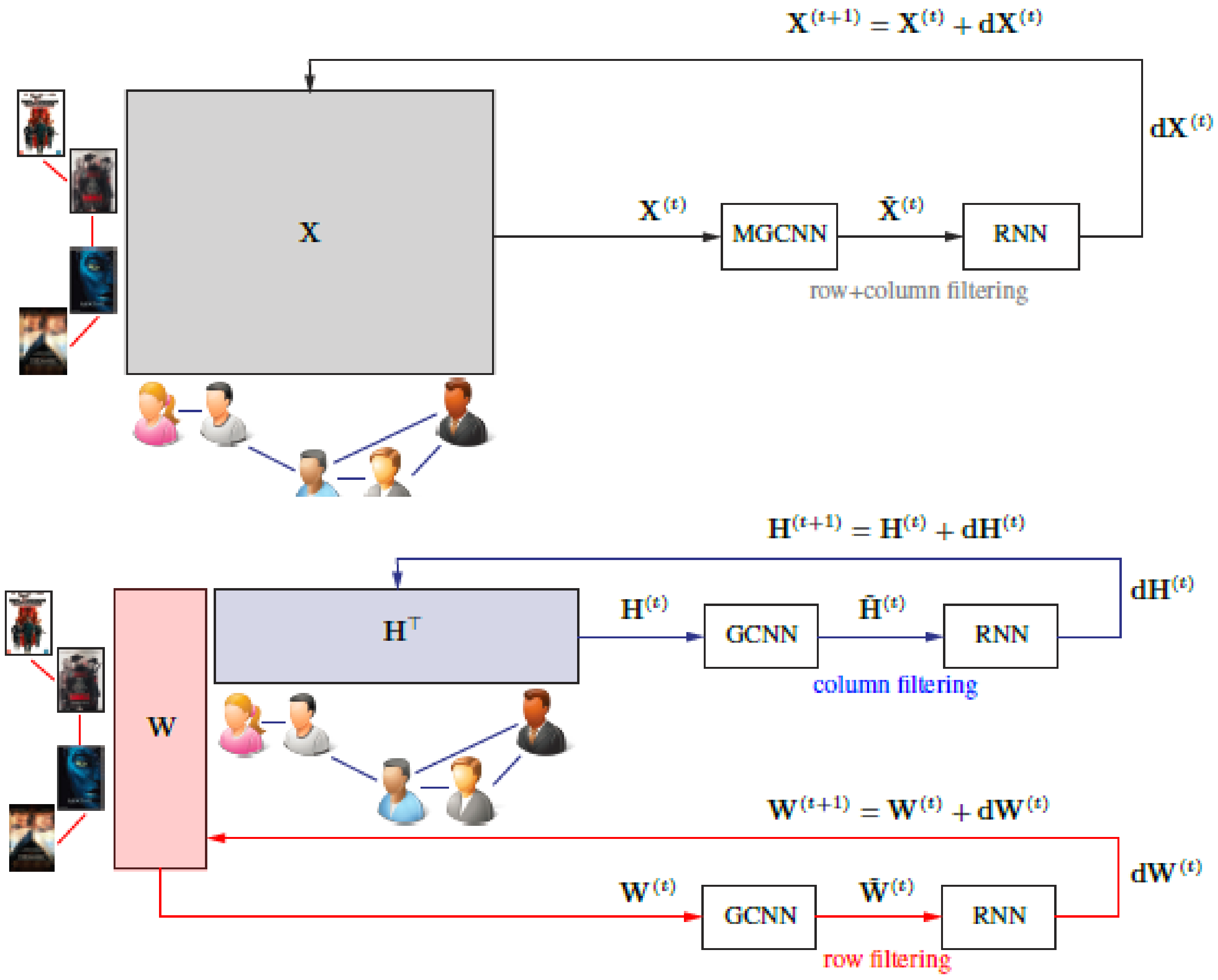
- Propose a new NN called EdgeConv
- Differentialable and Easy to be plugged into existing architectures
- EdgeConv:

$$x_i' = \underset{j:(i,j)\in\mathcal{E}}{\square} h_{\Theta}(x_i, x_j).$$



Wang Y, Sun Y, Liu Z, et al. Dynamic graph cnn for learning on point clouds[J]. arXiv preprint arXiv:1801.07829, 2018.

Application - Recommendation



Two forms of Fourier Transformations: columns and rows

$$\mathbf{X} \star \mathbf{Y} = \mathbf{\Phi}_r (\hat{\mathbf{X}} \circ \hat{\mathbf{Y}}) \mathbf{\Phi}_c^{\top}$$

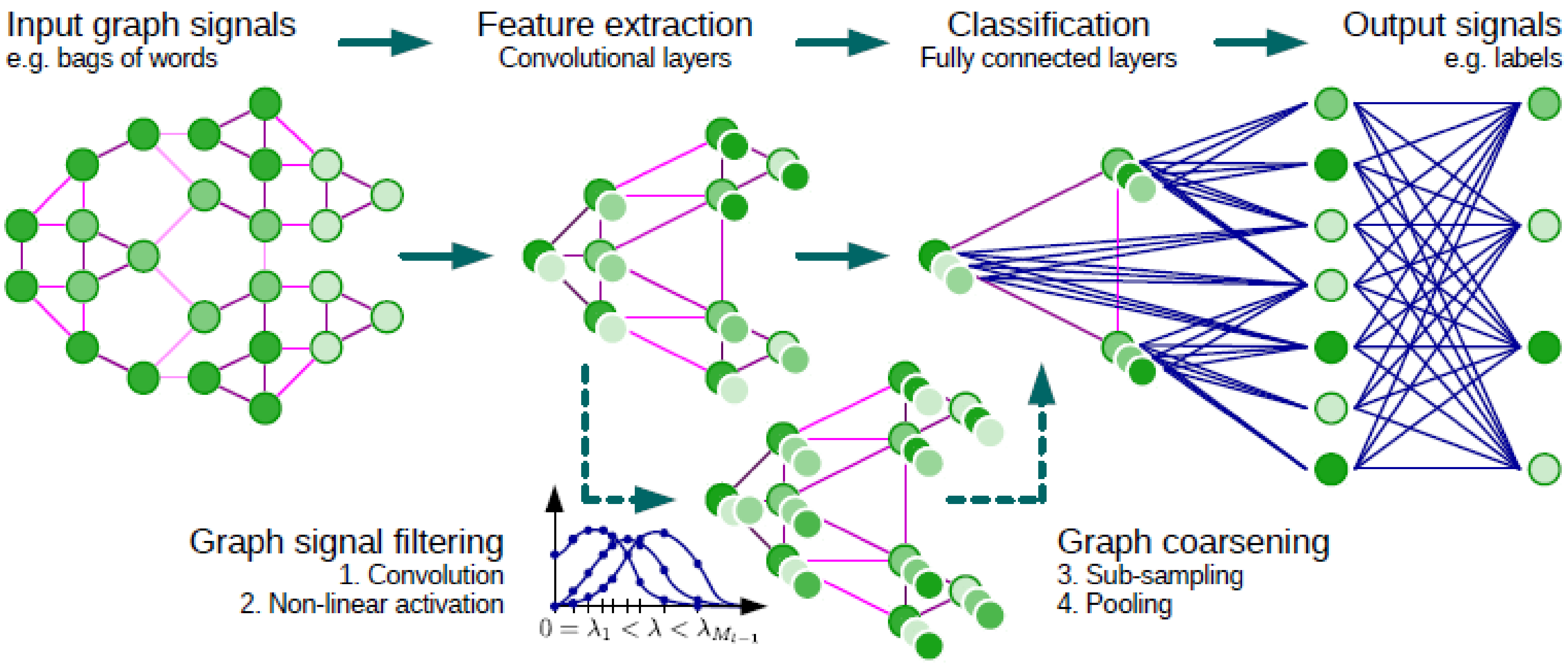
Matrix diffusion with RNN, to extract spatial features.

$$\mathbf{X} = \mathbf{W}\mathbf{H}^{\top}$$

Monti F, Bronstein M, Bresson X. Geometric matrix completion with recurrent multi-graph neural networks[C]//Advances in Neural Information Processing Systems. 2017: 3697-3707.

Application – Network analysis

Deal with social networks, such as citation networks.



Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering[C]//Advances in neural information processing systems. 2016: 3844-3852.

Questions and Future Work

- Generalization Problems
- Time-varying domains
- Hard to deal with directed-graphs
- Inefficient computation
- Generative Model

Resources

- Github:

  https://github.com/rusty1s/pytorch_geometric
  (Implementation of some latest geometric methods in pytorch)

  https://github.com/thunlp/GNNPapers
  (Latest GNN papers and models)

- Surveys:

  Bronstein M M, Bruna J, LeCun Y, et al. Geometric deep learning: going beyond euclidean data[J]. IEEE Signal Processing Magazine, 2017, 34(4): 18-42.

  Other GNN surveys provided by github links above..