ICML 2019

Disentangled Graph Convolutional Networks

Jianxin Ma, Peng Cui, Kun Kuang and Wenwu Zhu
Tsinghua University

Slides compiled by Mengzhou Li

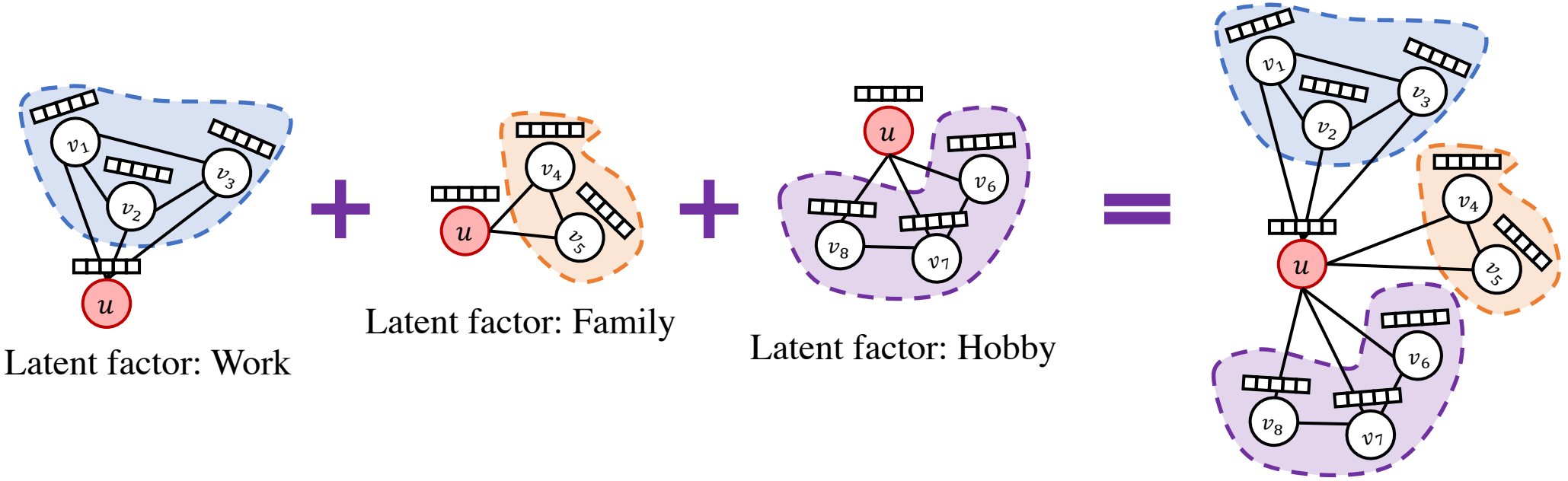
Contribution

- Present DisenGCN, a novel graph neural network that learns disentangled node representations.
- Propose neighborhood routing to infer the factor behind the formation of each edge, in a manner that is differentiable and supports inductive learning.
- Theoretically analyze the convergence properties of neighborhood routing, and empirically demonstrate the advantages of learning disentangled representations.

Following slides are borrowed from Jianxin Ma's Github

Motivation

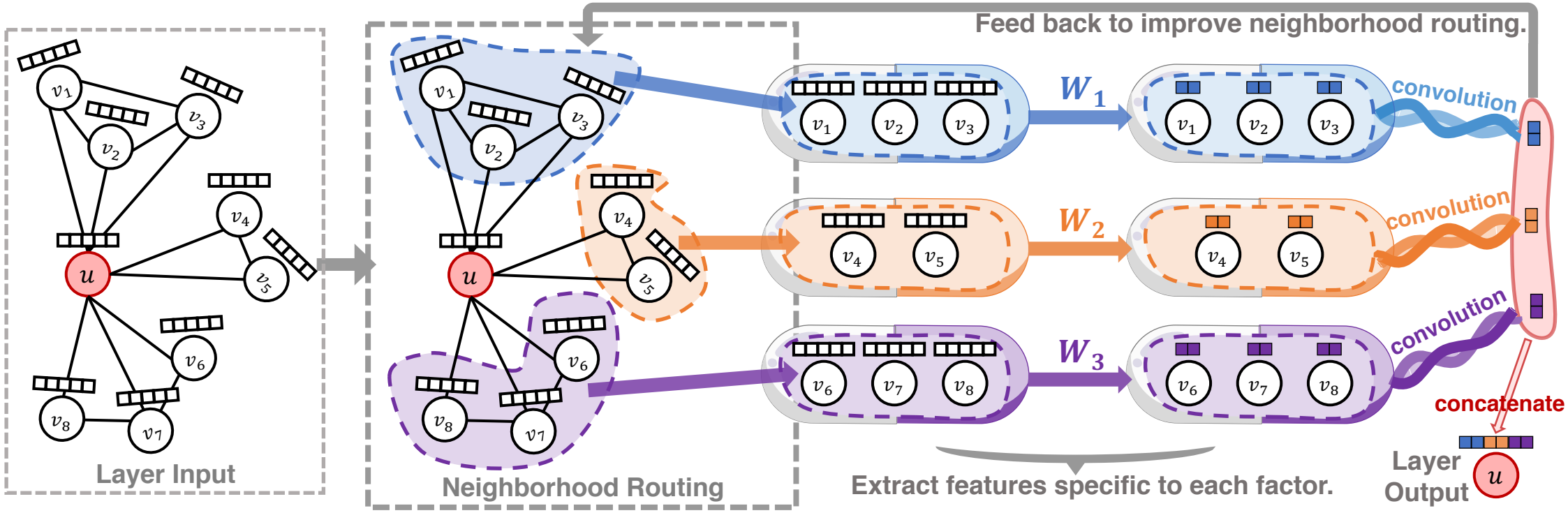
- The neighborhood of a node is formed due to *many latent factors*.



- Existing GCNs convolute the neighborhood as a whole.
  - They do not distinguish between the latent factors.
  - Their node representations are thus not *robust*, and hardly *interpretable.*

Disentangled GCNs

- *Disentangled representation learning* aims to identify and separate the underlying explanatory factors behind the observed data (Bengio et al., 2013).



- We identify the latent factors, and segment the neighborhood accordingly.
- Each segment is related with an isolated factor, and is convoluted separately.

Neighborhood Routing

The proposed layer takes in a set of node features $\{\boldsymbol{x}_u\} \cup \{\boldsymbol{x}_v : (v,u) \in G\}$, and outputs K representations $\{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_K\}$, where $\boldsymbol{c}_k$ describes the aspect related with factor k.

Phase I: To extract factor-specific features.

- For node $i \in \{u\} \cup \{v : (v,u) \in G\}$ and factor $k \in \{1,2,\ldots,K\}$, we compute:
  - $$\boldsymbol{z}_{i,k} = \frac{\sigma(\boldsymbol{W}_k^\top \boldsymbol{x}_i + \boldsymbol{b}_k)}{\left\|\sigma(\boldsymbol{W}_k^\top \boldsymbol{x}_i + \boldsymbol{b}_k)\right\|_2}.$$
- $\boldsymbol{z}_{i,k}$ stands for node i's aspect k. Each $\boldsymbol{z}_{i,k}$ may be incomplete or inaccurate. We are hence required to look at the whole picture by convoluting the neighborhood.

- ➢ It is differentiable, and therefore can be trained in an end-to-end manner.
- ➢ It supports real-time processing of new nodes, i.e., inductive learning.

Phase II (Neighborhood Routing): To identify the factor that causes the link between node u and a neighbor v.

- Initialize $\boldsymbol{c}_k \leftarrow \boldsymbol{z}_{u,k}$ for each factor k.
- Iteratively update $\boldsymbol{c}_k$ for $T \approx 5$ times:
  - $$p_{v,k} \leftarrow \frac{\exp\left(\boldsymbol{z}_{v,k}^\top \boldsymbol{c}_k / \tau\right)}{\sum_{k'} \exp\left(\boldsymbol{z}_{v,k'}^\top \boldsymbol{c}_{k'} / \tau\right)}$$
  - $$\boldsymbol{c}_k \leftarrow \frac{\boldsymbol{z}_{u,k} + \sum_{v:\,(v,u)\in G} p_{v,k}\, \boldsymbol{z}_{v,k}}{\left\|\boldsymbol{z}_{u,k} + \sum_{v:\,(v,u)\in G} p_{v,k}\, \boldsymbol{z}_{v,k}\right\|_2}$$
- $\boldsymbol{c}_k$ $(k = 1,2,\ldots,K)$ is a representation that describes the neighborhood's aspect k.

Intuitions & Theories

- **Intuitions:** The two intuitions behind neighborhood routing:
  - $p(\text{Factor } k \text{ is the one that causes the } \textbf{links between node } \boldsymbol{u} \textbf{ and a segment}) \propto$ The segment contains a large number of nodes that are similar w.r.t. aspect k.
  - $p(\text{Factor } k \text{ is the one that causes the } \textbf{link between node } \boldsymbol{u} \textbf{ and a neighbor}) \propto$ Node u and the neighbor are similar w.r.t. aspect k.

- **Theory:** Neighborhood routing is equivalent to an **expectation-maximization** (EM) algorithm that performs inference under a von Mises-Fisher subspace clustering model.
  - It finds one large cluster in each of the K subspaces, assuming that each neighbor belongs to one cluster.

Results: Semi-Supervised Node Classification

Table 1. Dataset statistics.

Dataset	Type	Nodes	Edges	Classes	Features	Multi-label
Citeseer	Citation network	3,327	4,732	6	3,703	No
Cora	Citation network	2,708	5,429	7	1,433	No
Pubmed	Citation network	19,717	44,338	3	500	No
Blogcatalog	Social network	10,312	333,983	39	-	Yes
PPI	Biological network	3,890	76,584	50	-	Yes
POS	Word co-occurrence	4,777	184,812	40	-	Yes

Nodes -> articles
Edges -> citations
Labels -> research areas

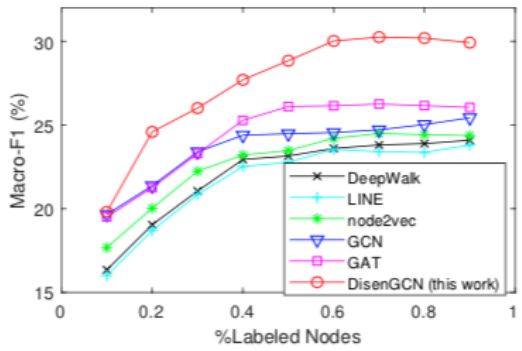
Each dataset contains only 20 labeled instances for class.

Table 2. Semi-supervised classification accuracies (%).

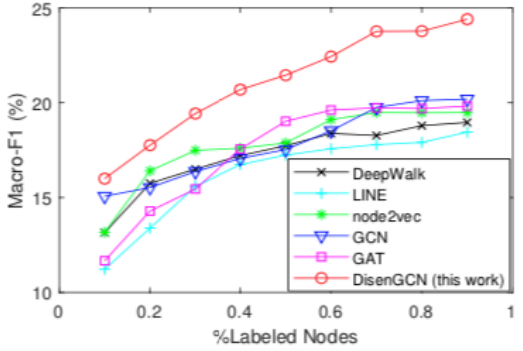
Method	Datasets		
	Cora	Citeseer	Pubmed
MLP	55.1	46.5	71.4
ManiReg (Belkin et al., 2006)	59.5	60.1	70.7
SemiEmb (Weston et al., 2012)	59.0	59.6	71.1
LP (Zhu et al., 2003)	68.0	45.3	63.0
DeepWalk (Perozzi et al., 2014)	67.2	43.2	65.3
ICA (Lu & Getoor, 2003)	75.1	69.1	73.9
Planetoid (Yang et al., 2016)	75.7	64.7	77.2
ChebNet (Defferrard et al., 2016)	81.2	69.8	74.4
GCN (Kipf & Welling, 2017)	81.5	70.3	79.0
MoNet (Monti et al., 2017)	81.7	-	78.8
GAT (Veličković et al., 2018)	83.0	72.5	79.0
DisenGCN (this work)	**83.7**	**73.4**	**80.5**

not suffer from the over-smoothing problem (deep GCN);
more resistant to over-fitting compared with a deep GAT (no extra parameters)

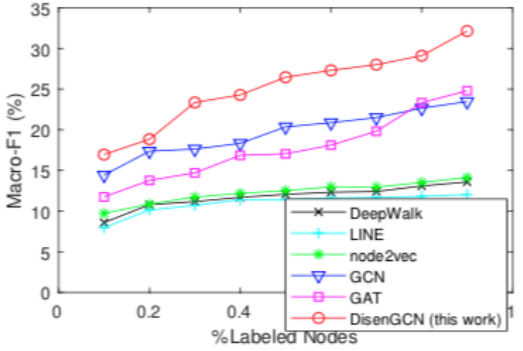
Results: Multi-label Node Classification



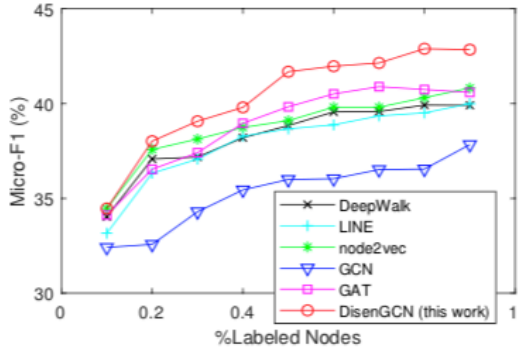
(a) Macro-F1(%), BlogCatalog.



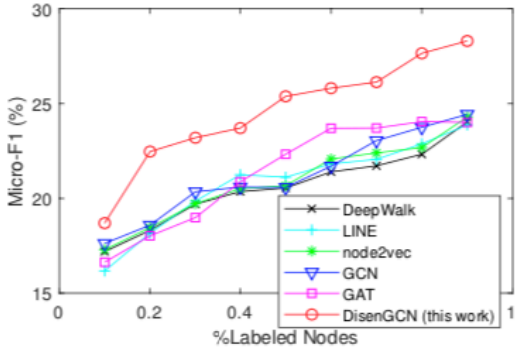
(c) Macro-F1(%), PPI.



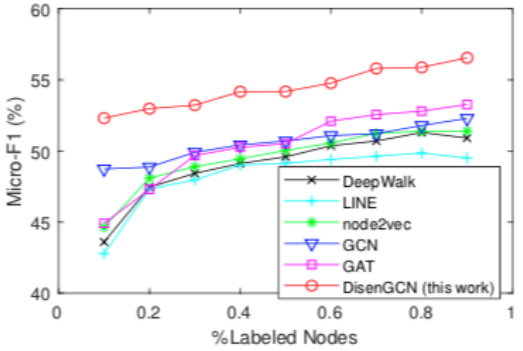
(e) Macro-F1(%), POS.



(b) Micro-F1(%), BlogCatalog.



(d) Micro-F1(%), PPI.



(f) Micro-F1(%), POS.

Figure 2. Macro-F1 and Micro-F1 scores on the multi-label classification tasks. Our approach consistently outperforms the best performing baselines by a large margin, reaching 10% to 20% relative improvement in most cases.

Thanks for your attention