INNER MONGOLIA UNIVERSITY

L²GC: Lorentzian Linear Graph Convolutional Networks For Node Classification

Qiuyu Liang, Weihua Wang *, Feilong Bao, Guanglai Gao
College of Computer Science, Inner Mongolia University

April 22th, 2024

Content

1.Introduction

We consider the problem of classifying nodes (such as documents) on the graph-structured data (such as citation network), where models need to learn information from their neighbours.

Graph Convolutional Network (GCN) is one of classification models that be able to aggregate and propagate the node information in graphs.

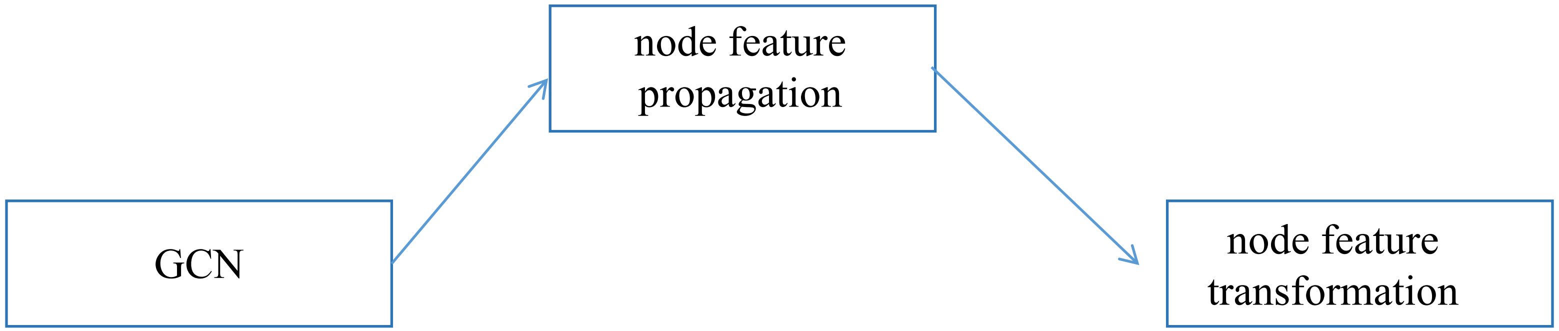
GCNs
GCN’ variants
...

achieved success →

text classification
relation extraction
social networks analysis
...

1.Introduction

In order to learn the features of each node, GCNs typically consist of two successive stages: node feature propagation and transformation.



1.Introduction

This multi-layer structure can obtain both local and global features naturally,

yet the computation complexity increased rapidly and further hinder their applications.

Therefore, some researchers attempted to improve the GCNs by eliminating the nonlinear active function between layers,

such as SGC and DGC, which are referred to as linear GCN model.

2.Motivation

- traditional linear GCN models mainly focus on improving the nodes feature propagation scheme, while pay fewer attention to the feature transformation stage.
- feature transformation in Euclidean space may be distorted when faced with this kind of data.

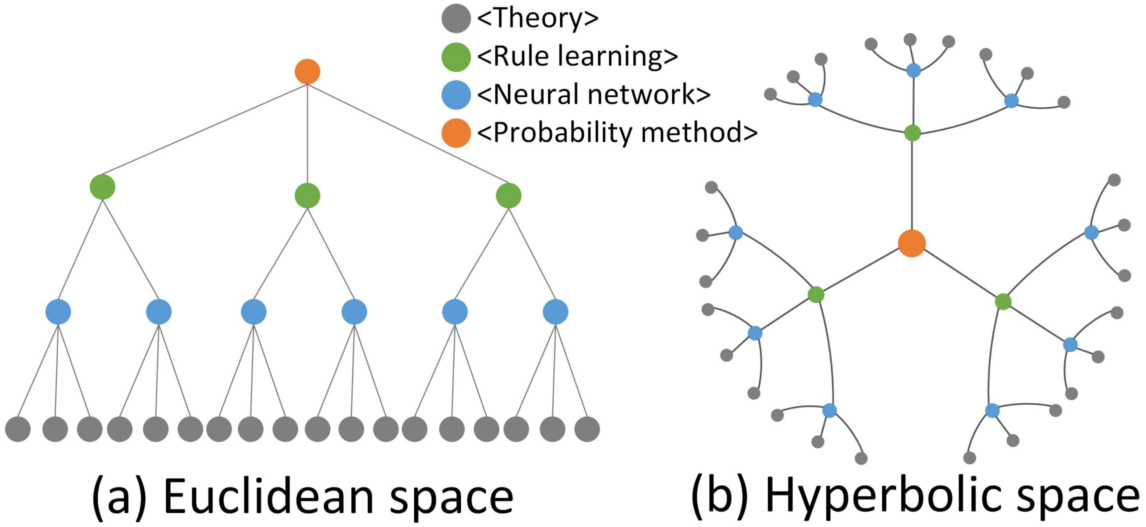


Fig. 1. Visualization of a tree-like hierarchical structure citation network in Euclidean and hyperbolic space.

3.Methodology

- We propose a novel Lorentzian Linear Graph Convolutional Networks (L^2GC) for classifying the tree-like graph node
- In hyperbolic space, we propose a translation operation on the head entity embedding in the relation embedding to obtain the support vector embeddings of the implicit relation embedding.

3.Methodology

Our framework mainly consists of three steps: **parameter-free neighborhood feature propagation** in Euclidean space, **Lorentzian linear feature transformation** in hyperbolic space and **graph node labels prediction** in Euclidean space.

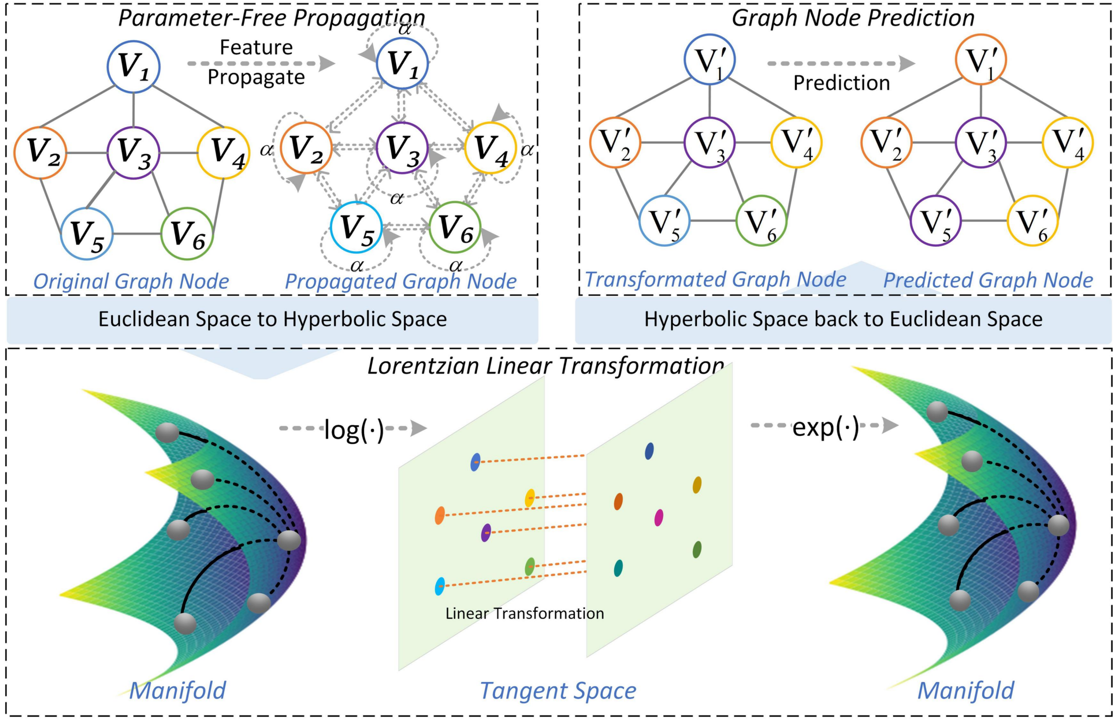


Fig. 2. The framework of Lorentzian Linear Graph Convolutional Networks for node Classification.

3.Methodology

Parameter-free neighborhood feature propagation.

The linear propagation matrix **P** can be written as:

$$\mathtt{P} = (\mathtt{D} + \mathtt{I})^{-1/2}(\mathtt{A} + \mathtt{I})(\mathtt{D} + \mathtt{I})^{-1/2}.$$

Then we precompute the information transfer between a node and its ***n***-power neighbors as follows:

$$\mathtt{H}^{(l+1)} = (1 - \alpha)\mathtt{P}\mathtt{H}^{(l)} + \alpha\mathtt{X},$$

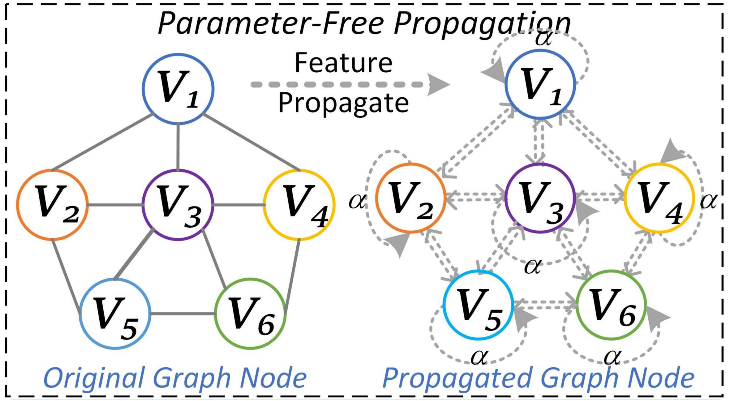


Fig. 3. Parameter-free neighborhood feature propagation.

3.Methodology

Lorentzian linear feature transformation.

After acquiring $\mathbf{H}^{(n)}$, we apply the exponential mapping to map the learned node features $\mathbf{H}^{(n)}$ into the hyperbolic space and then perform Lorentzian linear transformation.

This specific process is given by the following:

$$\mathrm{M}^{\otimes^k}\left(\mathtt{H}^{(n)}\right)=\exp_0^k\left(\hat{\mathrm{M}}\left(\log_0^k\left(\exp_0^k\left(\mathtt{H}^{(n)}\right)\right)\right)\right),$$

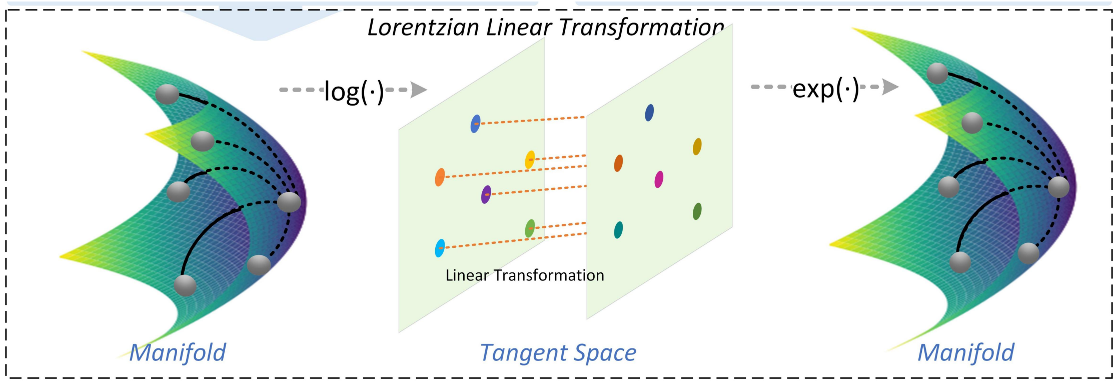


Fig. 4. Lorentzian linear feature transformation.

3.Methodology

Graph node labels prediction.

Finally, we use logarithm mapping to map the transformed node features back into Euclidean space for node prediction.

The process is as follows:

$$\mathtt{H}^{(n)'} = \log_0^k \left(\mathrm{M}^{\otimes^k} \left(\mathtt{H}^{(n)} \right) \right),$$

$$\hat{\mathtt{Y}} = \operatorname{argmax} \left(\mathtt{H}^{(n)'} \right),$$

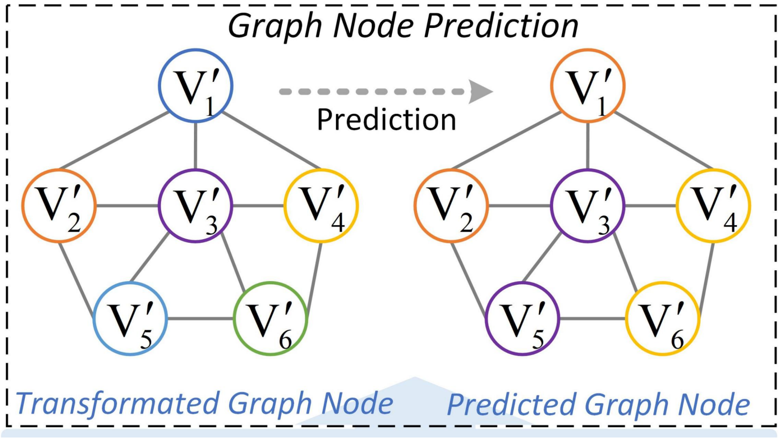


Fig. 5. Graph node labels prediction.

4.Experiments

Datasets

Dataset	Nodes	Edges	Label	Features	δ
Cora	$2,708$	$5,429$	7	$1,433$	11
Citeseer	$3,327$	$4,732$	6	$3,703$	4.5
PubMed	$19,717$	$44,338$	3	500	3.4
Disease	$1,044$	$1,043$	2	$1,000$	0
Airport	$3,188$	$18,631$	4	$3,188$	1

Baselines

Nonlinear models with Euclidean space: GCN, GAT, APPNP, GraphHeat, ElasticGNN, SCGNN

Linear models with Euclidean space: SGC, SIGN-linear, DGC, G^2CN, FLGC

Nonlinear models with hyperbolic space: HGCN, HAT, LGCN, HGCL, HYBONET

Linear model with hyperbolic space: Our method.

4.Experiments-Semi-supervised Node Classification

Test accuracy (%) of semi-supervised node classification on citation networks.

Space	Type	Method	Cora $\delta = 11$	Citeseer $\delta = 4.5$	PubMed $\delta = 3.4$
Euclidean	Nonlinear	GCN(Kipf and Welling, 2017)	81.5	70.3	79
		GAT(Veličković et al., 2018)	83.0	72.5 ± 0.7	79.0 ± 0.3
		APPNP(Gasteiger et al., 2018)	83.3	71.8	80.1
		GraphHeat(Xu et al., 2019)	83.7	72.5 ± 0.7	80.5
		ElasticGNN(Liu et al., 2021)	83.7 ± 0.2	72.2 ± 0.6	80.5 ± 0.1
		SCGNN(Liu et al., 2023)	**84.5** $\pm$ **0.3**	73.5 ± 0.5	80.8 ± 0.5
	Linear	SGC(Wu et al., 2019)	81.0 ± 0.0	71.9 ± 0.1	78.9 ± 0.0
		SIGN-linear(Frasca et al., 2020)	81.7	72.4	78.6
		DGC(Wang et al., 2021b)	83.3 ± 0.0	73.3 ± 0.1	80.3 ± 0.1
		G^2CN(Li et al., 2022)	82.7	73.8	80.4
		FLGC(Cai et al., 2023)	84.0 ± 0.0	73.2 ± 0.0	81.1 ± 0.0
Hyperbolic	Nonlinear	HGCN(Chami et al., 2019)	81.3 ± 0.6	70.9 ± 0.6	78.4 ± 0.4
		HAT(Zhang et al., 2021a)	83.1 ± 0.6	71.9 ± 0.6	78.6 ± 0.5
		LGCN(Zhang et al., 2021b)	83.3 ± 0.7	71.9 ± 0.7	78.6 ± 0.7
		HYBONET(Chen et al., 2022)	80.2 ± 1.3	-	78.0 ± 1.0
		HGCL(Liu et al., 2022)	82.3 ± 0.5	72.1 ± 0.6	79.14 ± 0.7
	Linear	**L^2GC**(ours)	82.4 ± 0.0	**74.7** $\pm$ **0.0**	**81.3** $\pm$ **0.0**

4.Experiments- Fully supervised Node Classification

Test accuracy (%) of fully supervised node.

Space	Method	Disease $\delta = 0$	Airport $\delta = 1$
$\mathbb{E}$	GCN(2017)	69.7 ± 0.4	81.4 ± 0.6
	GAT(2018)	70.4 ± 0.4	81.5 ± 0.3
	SGC(2019)	69.5 ± 0.2	80.6 ± 0.1
	SCGNN(2023)	85.3 ± 0.4	-
$\mathbb{H}$	HGCN(2019)	82.8 ± 0.8	90.6 ± 0.2
	HAT(2021a)	83.6 ± 0.9	-
	LGCN(2021b)	84.4 ± 0.8	90.9 ± 1.7
	HGCL(2022)	93.4 ± 0.8	92.3 ± 1.0
	HYBONET(2022)	$\mathbf{96.0 \pm 1.0}$	90.9 ± 1.4
	L^2GC(ours)	94.4 ± 0.1	$\mathbf{94.0 \pm 0.1}$

4.Experiments-Ablation

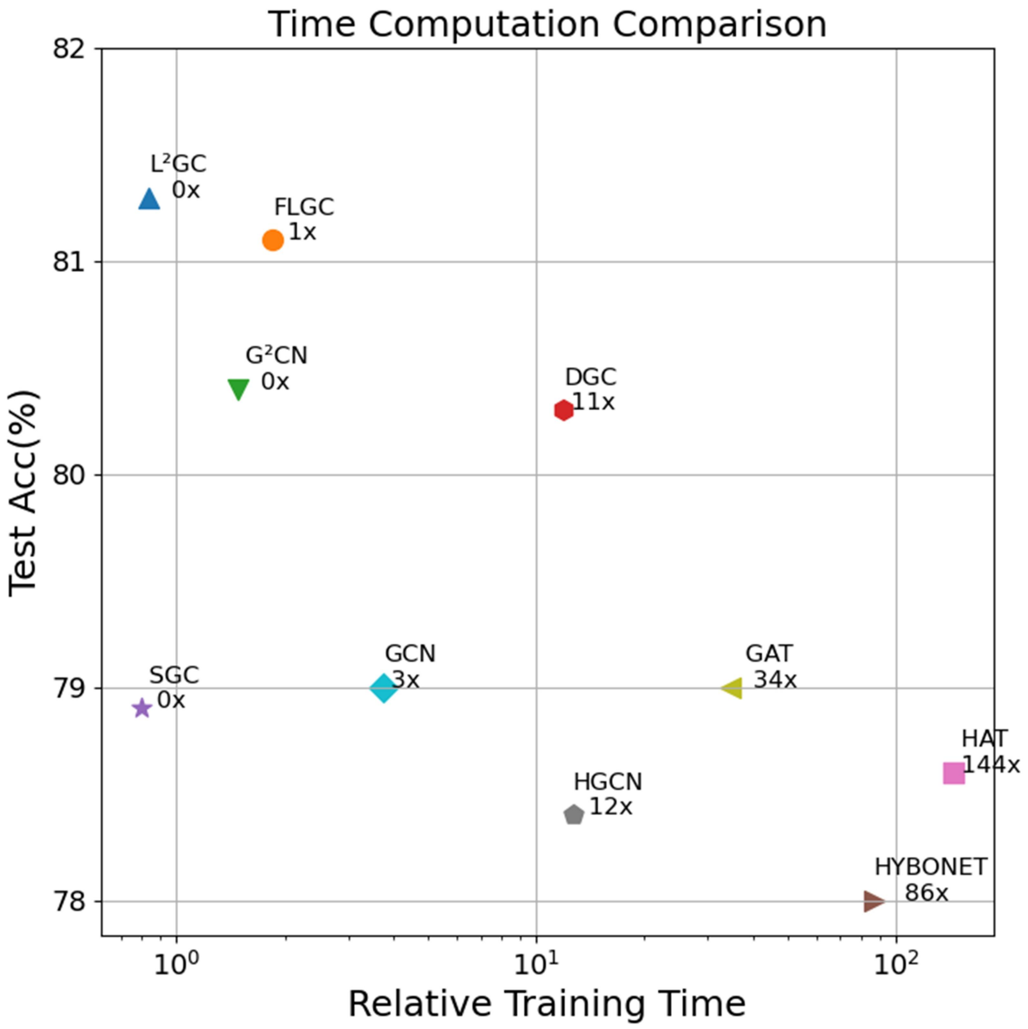
Variant I : L^2GC with Personalized Propagation scheme and without Lorentz model, which indicates the features transformation stage in Euclidean space.

Variant II : L^2GC without Personalized Propagation scheme and with Lorentz model, we use the SGC propagation scheme in the feature propagation stage.

The ablation experiments of **L²**GC.

	Cora $\delta = 11$	**Citeseer** $\delta = 4.5$	**PubMed** $\delta = 3.4$
$Variant\ I$	**83.1**	73.0	79.4
$Variant\ II$	80.2	73.6	79.7
L^2GC	82.4	**74.7**	**81.3**

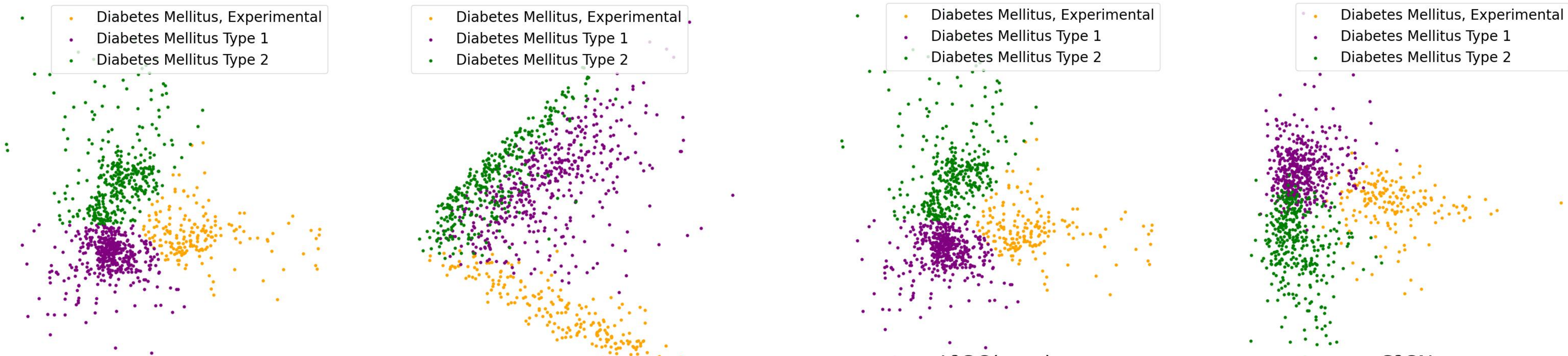
4.Experiments-Efficiency analysis



4.Experiments-Parameter analysis

Model	HYBONET	L^2GC(ours)
Cora	23356	8604 ($\downarrow$ 63.16%)
Citeseer	59659	18520 ($\downarrow$ 68.95%)
PubMed	8360	1002 ($\downarrow$ 88.01%)

4.Experiments-Visualisation



L^{2}GC(ours) HYBONET L^{2}GC(ours) G^{2}CN

5.Conclusion

In this paper, we propose a novel Lorentzian Linear Graph Convolutional Networks framework for node classification based on hyperbolic space.

Our work is the first generalization of the linear GCN model to hyperbolic space, which capturing of the hierarchical structure in the data.

Our approach not only leverages the strengths of the linear model, but also integrates the properties of hyperbolic spaces to achieve new SOTA results on the Citeseer and PubMed datasets.

THANKS