# Composition-based Multi-Relational Graph Convolutional Networks





Shikhar Vashishth*[1,2]
svashish@cs.cmu.edu

<u>Soumya Sanyal</u>*[1]
soumyasanyal@iisc.ac.in

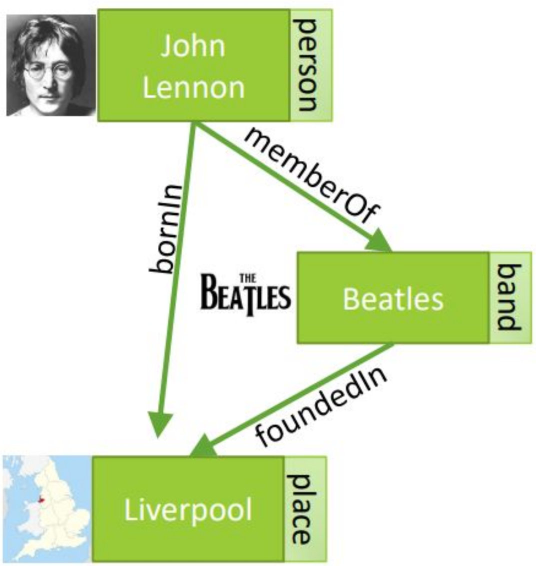
Vikram Nitin[1,3]
vikram.nitin@columbia.edu

Partha Talukdar[1]
ppt@iisc.ac.in

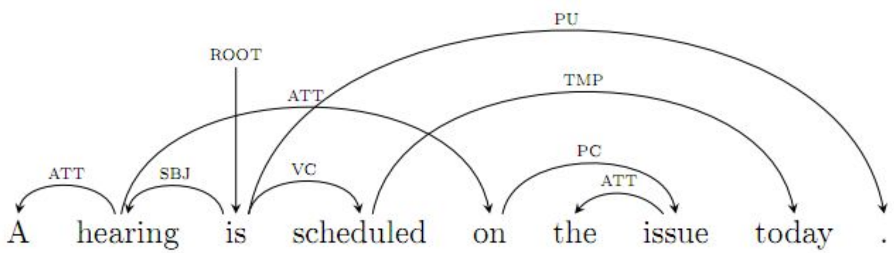
**[1]Indian Institute of Science, [2]Carnegie Mellon University, [3]Columbia University**

# Multi-relational Graphs

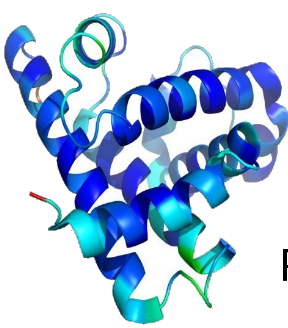
- Graphs with **directed-labeled edges**
- Multi-relational graphs are pervasive, examples include...




Knowledge Graphs

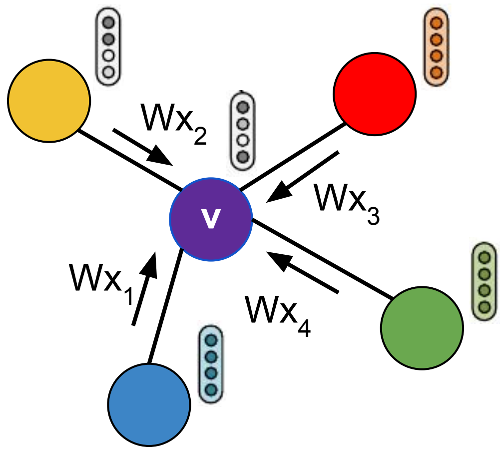



Dependency Parse



Proteins

# Graph Convolutional Networks (GCNs)




GCN First-order approximation

(Kipf et. al. 2016)

$$h_v = f\left(\frac{1}{|\mathcal{N}(v)|}\sum_{u \in \mathcal{N}(v)} W x_u + b\right), \quad \forall v \in \mathcal{V}.$$

- Most GCN formulations are for simple undirected graphs
- Naive extension of GCNs to Multi-relational graphs using relation-specific filter matrix (W)
  - Suffers from overparameterization

# Existing Multi-Relational GCN models

| **Methods** | Node Embeddings | Directions | Relations | Relation Embeddings |
|---|---|---|---|---|
| GCN Kipf & Welling (2016) | ✓ | | | |
| Directed-GCN Marcheggiani & Titov (2017) | ✓ | ✓ | | |
| Weighted-GCN Shang et al. (2019) | ✓ | | ✓ | |
| Relational-GCN Schlichtkrull et al. (2017) | ✓ | ✓ | ✓ | |

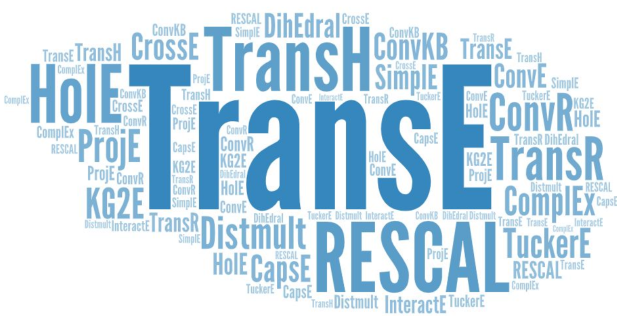
- **Directed-GCN:** Utilizes direction-specific filter matrix
- **Weighted-GCN:** Learns a scalar weight for each relation
- **Relational-GCN:** Relation-specific filters in terms of basis matrices

Although solve overparameterization to different degrees of granularity, none of them learn relation embeddings

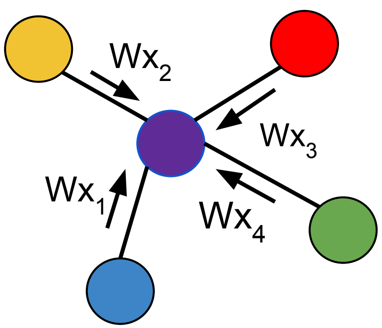
# Motivation

- Extensive research done on **embedding Knowledge Graphs** where representations of both nodes and relations are jointly learned.

- Can we develop a **GCN framework** that can leverage the advances in KGE approaches to:
  - Learn both node and relation embeddings
  - Solve the issue of overparameterization




KG Embedding methods



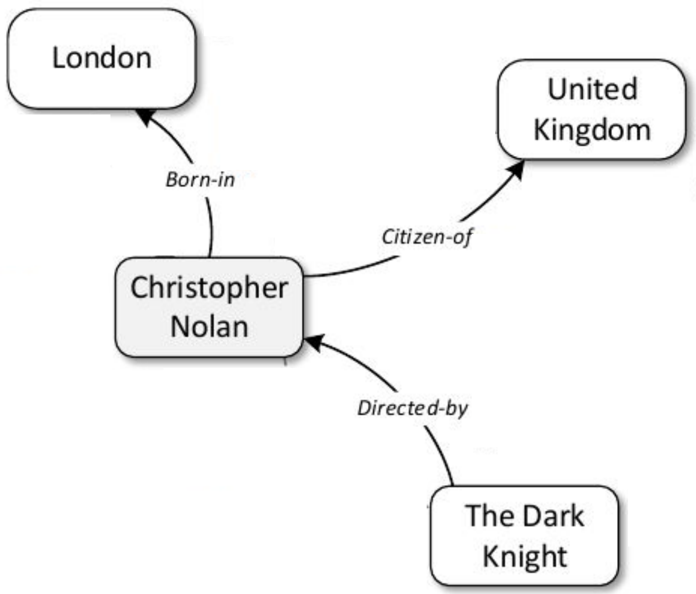

Graph ConvNets

# Contributions

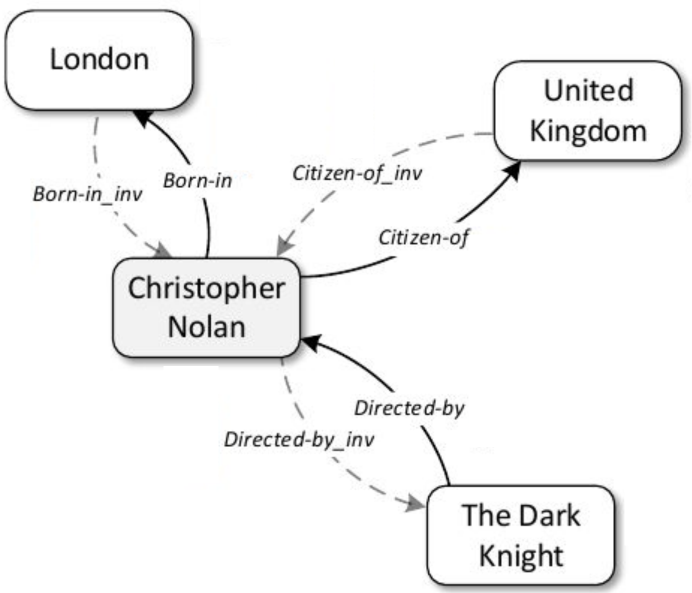
- We propose **CompGCN**, a novel framework for incorporating multi-relational information in GCNs which leverages a variety of composition operations from KG embedding techniques.
- Unlike previous GCN methods, CompGCN jointly learns to embed both nodes and relations in the graph

| **Methods** | Node Embeddings | Directions | Relations | Relation Embeddings |
|---|---|---|---|---|
| GCN Kipf & Welling (2016) | ✓ | | | |
| Directed-GCN Marcheggiani & Titov (2017) | ✓ | ✓ | | |
| Weighted-GCN Shang et al. (2019) | ✓ | | ✓ | |
| Relational-GCN Schlichtkrull et al. (2017) | ✓ | ✓ | ✓ | |
| CompGCN (Proposed Method) | ✓ | ✓ | ✓ | ✓ |

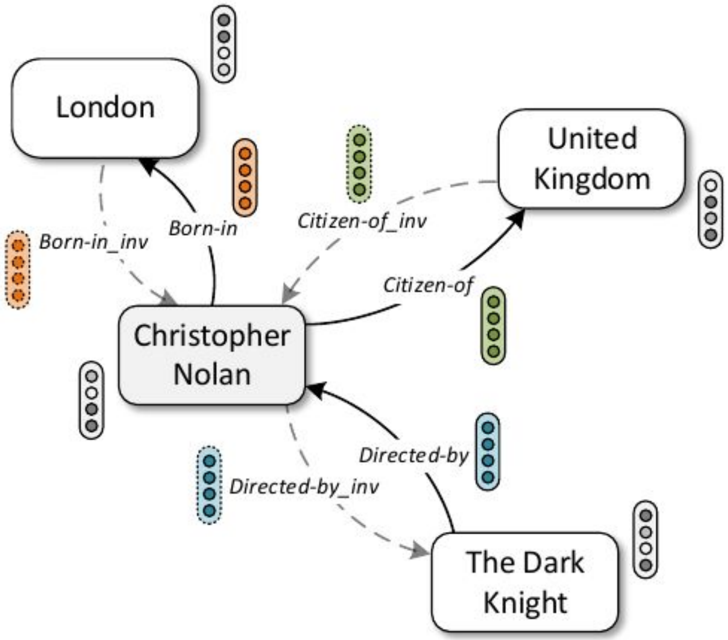
# CompGCN: Overview



# CompGCN: Overview

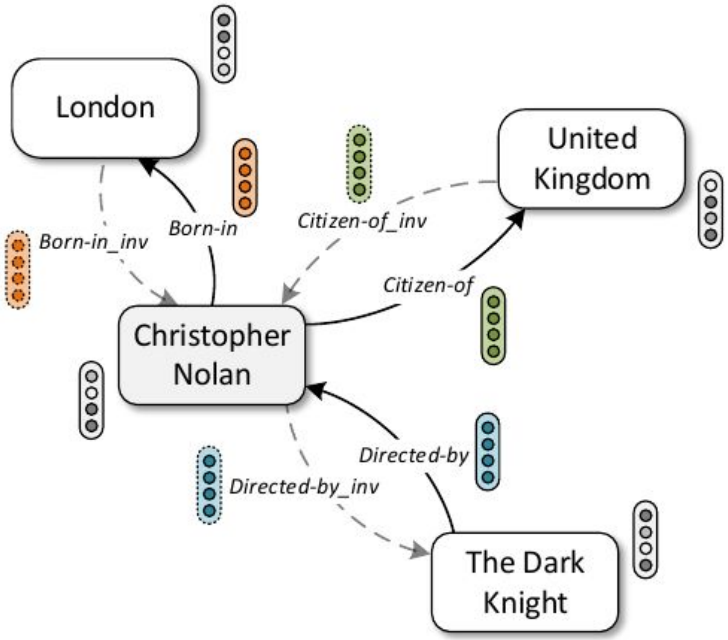


# CompGCN: Overview

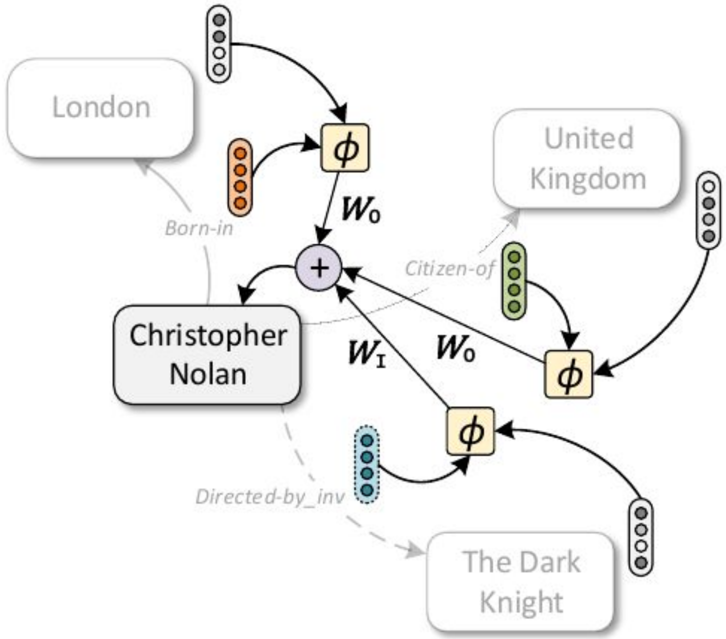



**Relational Graph with Embeddings**

# CompGCN: Overview



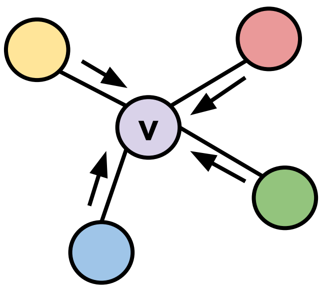

**Relational Graph with Embeddings**




**CompGCN Update**

# CompGCN: Update Equation

**Node Update:**




*non-linearity*

*Composition Operation*

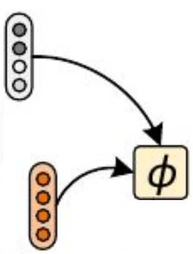
$$\boldsymbol{h}_v^{k+1} = \boxed{f}\left(\boxed{\sum_{(u,r)\in\mathcal{N}(v)}} \boxed{\boldsymbol{W}_{g(r)}^k} \boxed{\phi(\boldsymbol{h}_u^k, \boldsymbol{h}_r^k)}\right)$$

*Summation over neighborhood of v*

$$\boldsymbol{W}_{g(r)} = \begin{cases} \boldsymbol{W}_O, & r \in \mathcal{R} \\ \boldsymbol{W}_I, & r \in \mathcal{R}_{inv} \\ \boldsymbol{W}_S, & r = \top \end{cases}$$

Original edges
Inverse Edges
Self Loops

**Composition Operation:**




$$\phi(\boldsymbol{h}_u^k, \boldsymbol{h}_r^k) = \begin{cases} \boldsymbol{e}_s - \boldsymbol{e}_r & \text{Subtraction (TransE)} \\ \boldsymbol{e}_s * \boldsymbol{e}_r & \text{Multiplication (DistMult)} \\ \boldsymbol{e}_s \star \boldsymbol{e}_r & \text{Circular-correlation (HolE)} \end{cases}$$

**Relation Update:**

$$\boldsymbol{h}_r^{k+1} = \boldsymbol{W}_{rel}^k \, \boldsymbol{h}_r^k$$

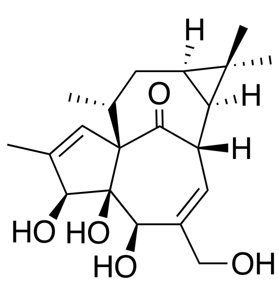
# Evaluation Tasks

- ## Link Prediction in Knowledge Graph


Knowledge Graph

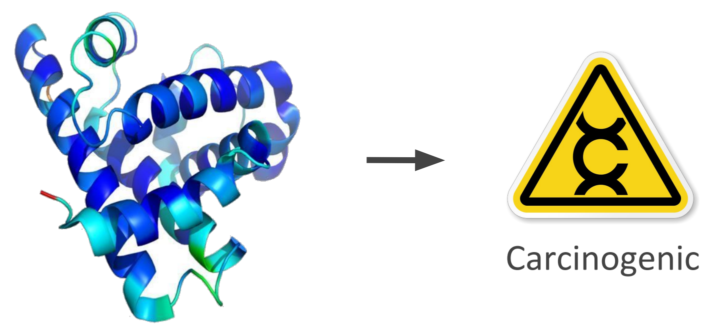
Inferring missing links

- ## Node Classification




Functional Group Classification

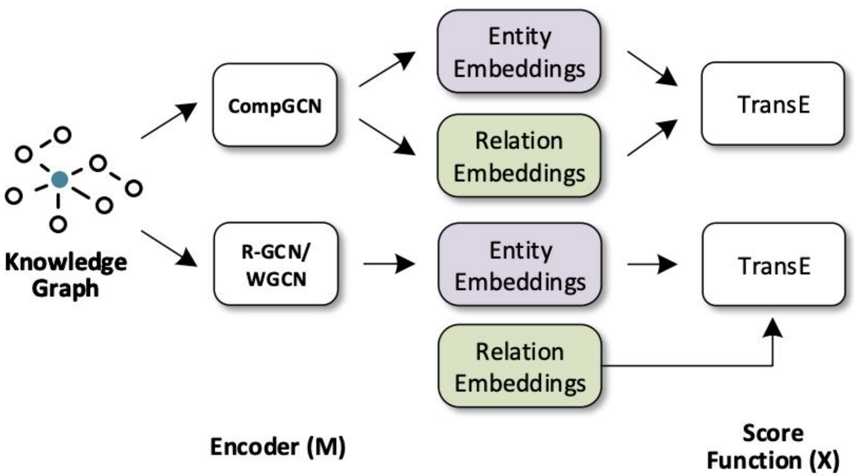
- ## Graph Classification




Molecule Classification

# CompGCN: Link Prediction Results

- Effect of different GCN models and composition operators




| Scoring Function (=X) → | ConvE | | |
|---|---|---|---|
| Methods ↓ | MRR | MR | H@10 |
| X | 0.325 | 244 | 0.501 |
| X + D-GCN | 0.344 | 200 | 0.524 |
| X + R-GCN | 0.342 | 197 | 0.524 |
| X + W-GCN | 0.344 | 201 | 0.525 |
| X + CompGCN (Sub) | 0.352 | 199 | 0.530 |
| X + CompGCN (Mult) | 0.353 | 216 | 0.532 |
| X + CompGCN (Corr) | **0.355** | 197 | **0.535** |
| X + CompGCN ($\mathcal{B} = 50$) | 0.350 | **193** | 0.530 |

**ConvE + CompGCN(Corr)** gives best performance across all settings.

# CompGCN: Link Prediction Results

- Performance on Link Prediction

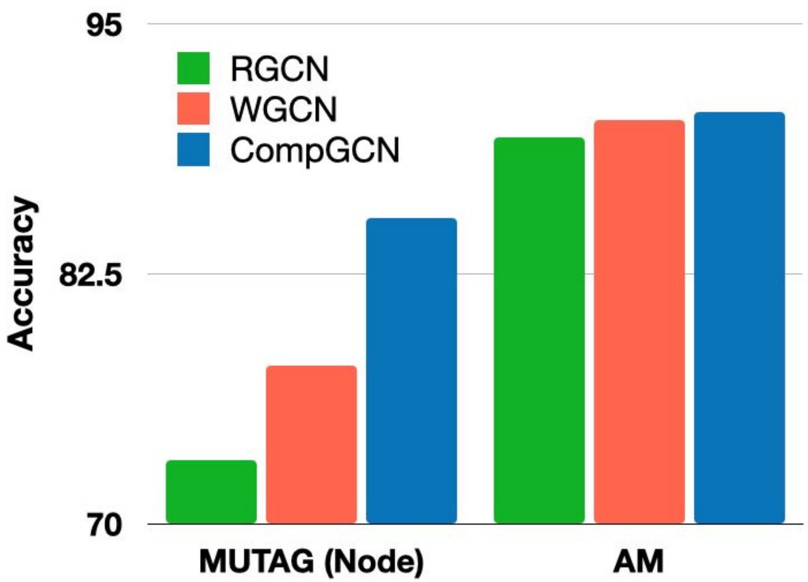
| | FB15k-237 | | | | WN18RR | | | |
|---|---|---|---|---|---|---|---|---|
| | MRR | H@10 | H@3 | H@1 | MRR | H@10 | H@3 | H@1 |
| R-GCN | .248 | .417 | - | .151 | - | - | - | - |
| ConvE | .325 | .501 | .356 | .237 | .43 | .52 | .44 | .40 |
| SACN | .35 | **.54** | **.39** | .26 | .47 | .54 | .48 | .43 |
| RotatE | .338 | .533 | .375 | .241 | .476 | **.571** | .492 | .428 |
| CompGCN | **.355** | **.535** | **.390** | **.264** | **.479** | .546 | **.494** | **.443** |

**CompGCN** provides a consistent improvement across all the datasets.

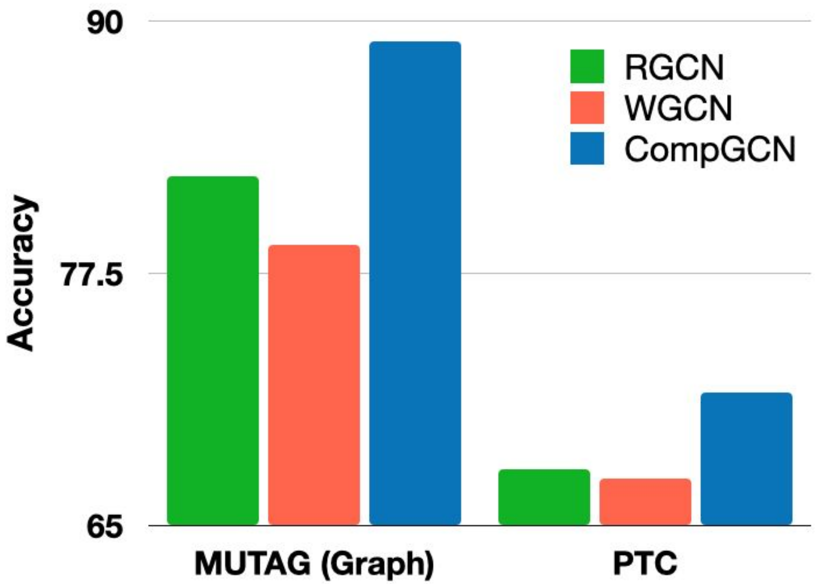
# CompGCN: Results

- Performance on Node Classification and Graph Classification

Node classification Performance



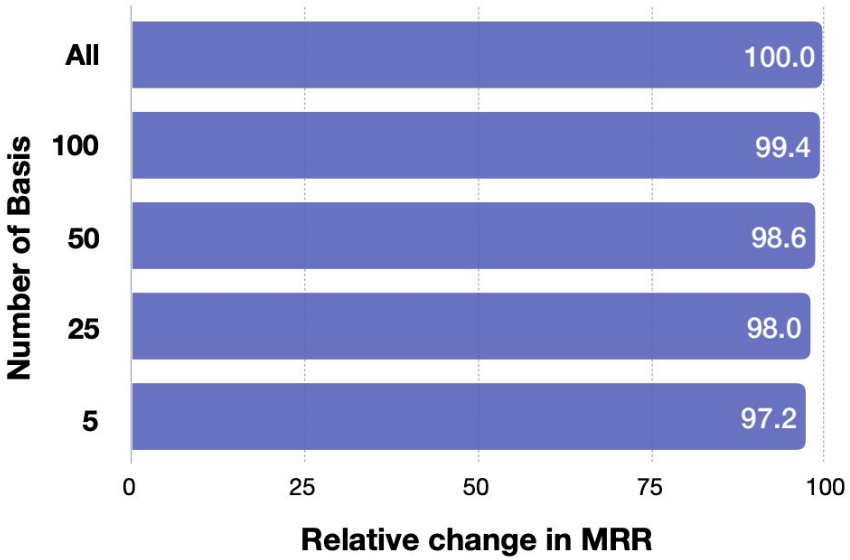

Graph classification Performance

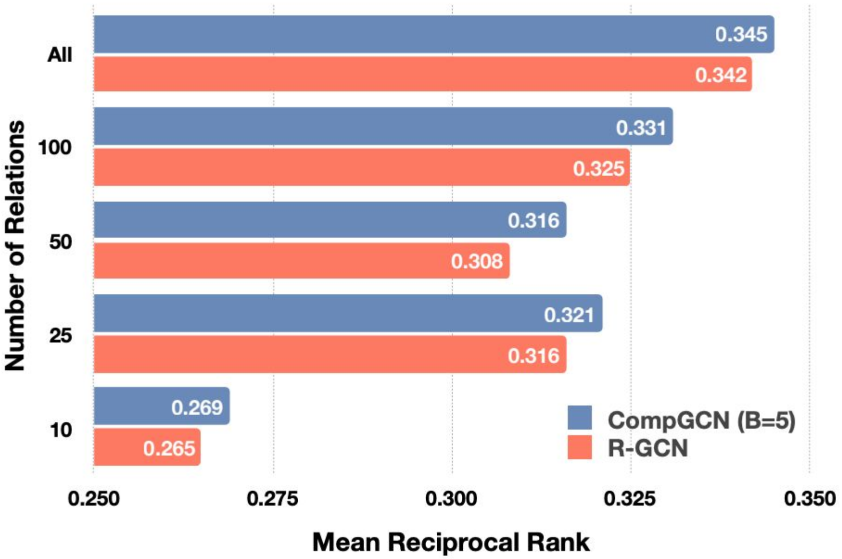



**CompGCN** outperforms or performs comparably to existing baselines.

# CompGCN: Scalability

- Effect of number of relation basis vectors and relations on FB15k-237




**COMPGCN** outperforms RGCN even with limited parameters.

# Conclusion

- Multi-relational graphs are prevalent in real-world problems.

- Current GCN approaches mainly focus on simple undirected graphs.

- We propose **CompGCN**, a parameter efficient method for embedding both nodes and relation types.

- We demonstrate the effectiveness of CompGCN for link prediction, node and graph classification tasks.

**Paper Link:**
[Composition-Based Multi-Relational Graph Convolutional Networks]

# Thank you!

**Source Code:**

PyTorch geometric

*based implementation*

github.com/malllabiisc/CompGCN

**Research Supported by:**

MHRD
Govt. of India

सत्यमेव जयते