GNN-based Biomedical Knowledge Graph Mining in Drug Development

Maria Wyrzykowska
Seminar: Data Mining - Clustering and Classification

Chapter 24
GNN-based Biomedical Knowledge Graph Mining in Drug Development

Chang Su, Yu Hou, Fei Wang

https://graph-neural-networks.github.io/

Agenda

1. **Data**
   - Characteristics & challenges
   - Biomedical Knowledge Graphs
2. **Inference on Knowledge Graphs**
   - Conventional inference techniques
   - GNN-based inference techniques
3. **Real-life applications**
   - Drug repurposing
   - Limitations & future directions

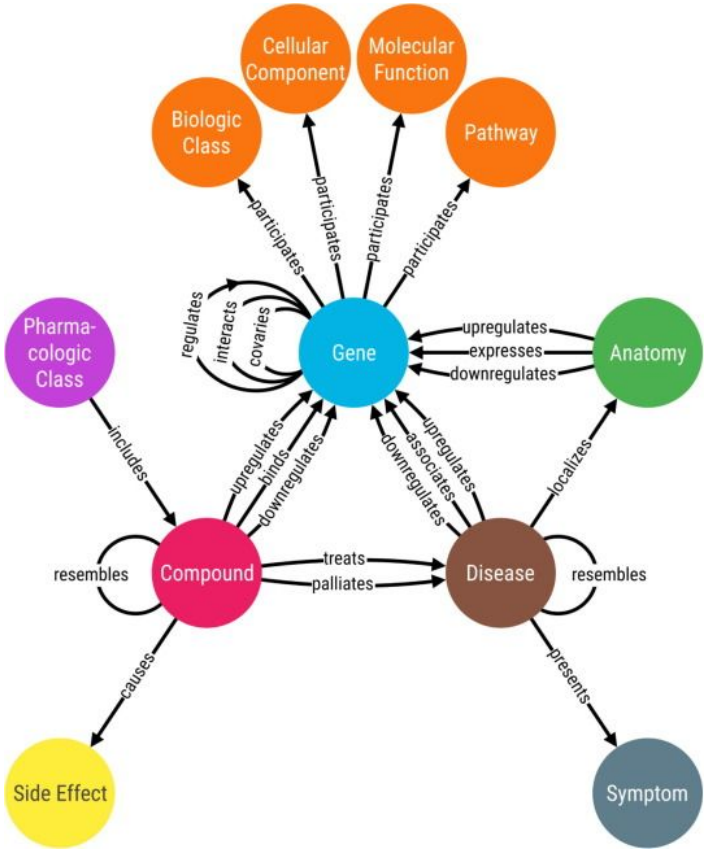
Data

Biomedical data

- **a lot** of it…
- …but is usually buried in the literature
- **heterogenous**: genes, diseases, drugs, interactions
- **relational**

knowledge retrieval & organization is difficult

Solution: using graphs!



- set of **<head, relation, tail>** tuples
- created by **extraction** and **integration** of data from other sources
- **examples:** Hetionet, Drug Repurposing Knowledge Graphs and many others
- usage: prior knowledge to different models, **generation of hypotheses**

(Biomedical) Knowledge graphs

Inference on Knowledge Graphs

Inference on Knowledge Graphs

Important attributes to take into consideration:

- local and global structure properties
- heterogeneity of entities and relations

Standard pipeline:

1. Learning embeddings
2. Performing downstream tasks e.g. link prediction

Approaches:

1. Conventional: semantic matching, distance models, meta-path-based, CNN
2. GNN-based: GCN, GAT

Common notation

Entity	Symbol
Entities (diseases, drugs, genes)	E
Entity i (disease, drug, gene)	e_i
Relation k (“cures”, “causes”)	r_k
Embedding of entity i	$h_i \in \mathcal{R}^n$
Embedding of relation k	$g_k \in \mathcal{R}^n$

Conventional KG inference: semantic matching models

Based on idea that entities are similar if connected to similar entities via similar relations.

Example: **RESCAL**

$$f(e_i, r_k, e_j) = \mathbf{h}_i^\top M_k \mathbf{h}_j$$

where:

M_k - embedding matrix, $M_k \in \mathcal{R}^{n \times n}$

Loss used for training is standard RMSE + regularization.

[A Three-Way Model for Collective Learning on Multi-Relational Data]

Conventional KG inference: translational distance model

Based on idea that relation can be considered as a translation from head entity to tail entity in the embedding space.

Example: **TransE**

$$f(e_i, r_k, e_j) = ||\mathbf{h_i} + \mathbf{g_k} - \mathbf{h_j}||$$

where:

$||\ ||_2$ - Euclidean norm

[Translating Embeddings for Modeling Multi-relational Data]

TransE: training

Loss function:

$$L = \sum_{(e_i, r_k, e_j) \in S} \sum_{(e'_i, r_k, e'_j) \in S'} [\gamma + d(h_i + g_k, h_j) - d(h'_i + g_k, h'_j)]_+$$

where:

S' - corrupted triplets

$$S' = \{(e'_i, r_k, e_j) | e'_i \in E\} \cup \{(e_i, r_k, e'_j) | e'_j \in E\}$$

γ - margin

d - dissimilarity measure (e.g. L1 or L2 norm)

Translating Embeddings for Modeling Multi-relational Data

TransE: modifications

TransE does well for 1-to-1 relations, but does not manage to model N-to-1, 1-to-N and N-to-N relations. Many modifications try to alleviate this problem:

- TransH
- TransR
- TransD
- TranSparse
- TransF
- …

Most are based on idea to project the low-dimensional embeddings to hyperplanes, different for each relation.

Conventional KG inference: meta-path-based approaches

Issue of semantic & distance models is that they focus on one-hop neighbourhoods.
The meta-path based models aim at capturing both local and global structure properties.

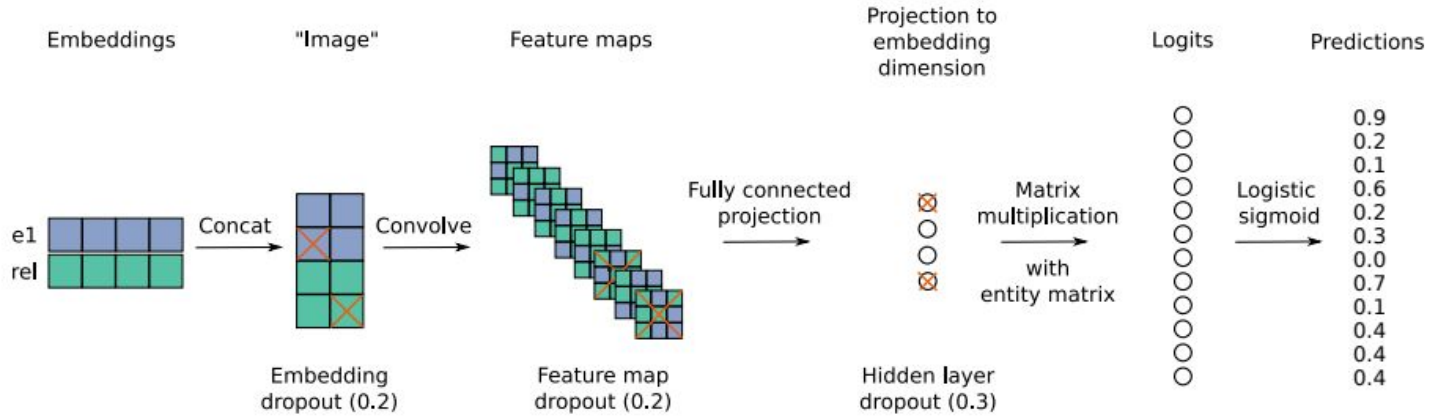
$$a_1 \xrightarrow{b_1} a_2 \xrightarrow{b_2} \ldots \xrightarrow{b_{l-1}} a_l$$

- **Heterogeneous Information Network Embedding** (HINE) - minimizing the difference between meta-path-based proximity and expected proximity in the embedding space
- **metapath2vec** - random walks in the graph are treated as sentences and SkipGram with negative sampling is used to learn embeddings

Heterogeneous Information Network Embedding for Meta Path based Proximity,
metapath2vec: Scalable Representation Learning for Heterogeneous Networks

Conventional KG inference: CNN models

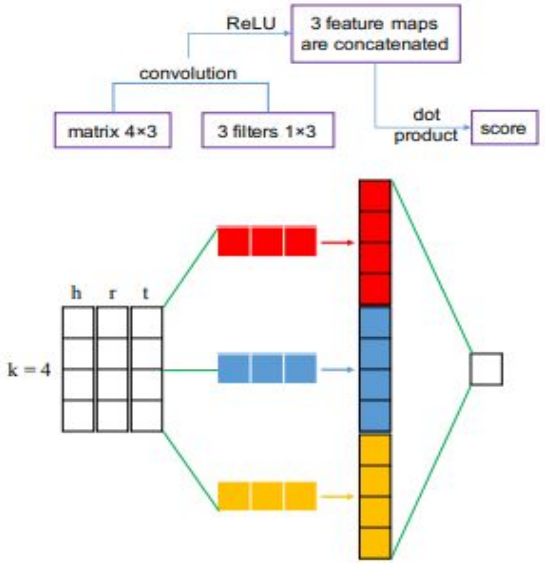
ConvE



[Convolutional 2D Knowledge Graph Embeddings]

Conventional KG inference: CNN models

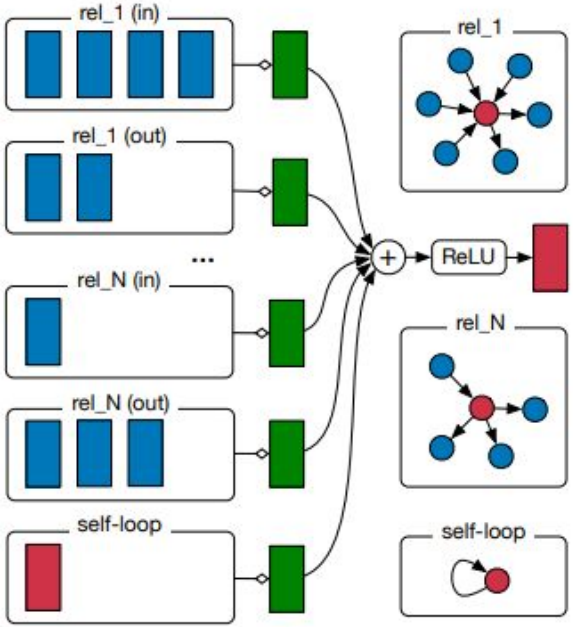
ConvKB



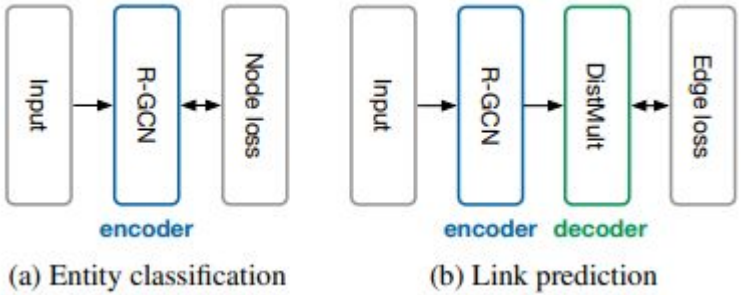
A Novel Embedding Model for Knowledge Base Completion Based on Convolutional Neural Network

GNN-based KG inference: GCN

Relational GCN (R-GCN)



$$\mathbf{h}_i^{(l+1)} = \sigma\left(\sum_{r_k \in \mathbb{R}} \sum_{j \in \mathcal{N}_i^k} \frac{1}{c_{i,k}} W_k^{(l)} \mathbf{h}_j^{(l)} + W_0^{(l)} \mathbf{h}_i^{(l)}\right)$$



(a) Entity classification (b) Link prediction

[Modeling Relational Data with Graph Convolutional Networks]

GNN-based KG inference: GCN

TransGCN combines GCN & translational distance models (e.g. TransE) to learn both entities and relations embeddings.

At layer l:

1. Message propagation:

$$m_i^{(l+1)} = \frac{1}{c_i} W_0^{(l)} \left(\sum_{(e_j, r_k, e_i) \in T_{in}(e_i)} (h_j^{(l)} \circ g_k^{(l)}) + \sum_{(e_i, r_k, e_j) \in T_{out}(e_i)} (h_j^{(l)} \star g_k^{(l)}) \right)$$

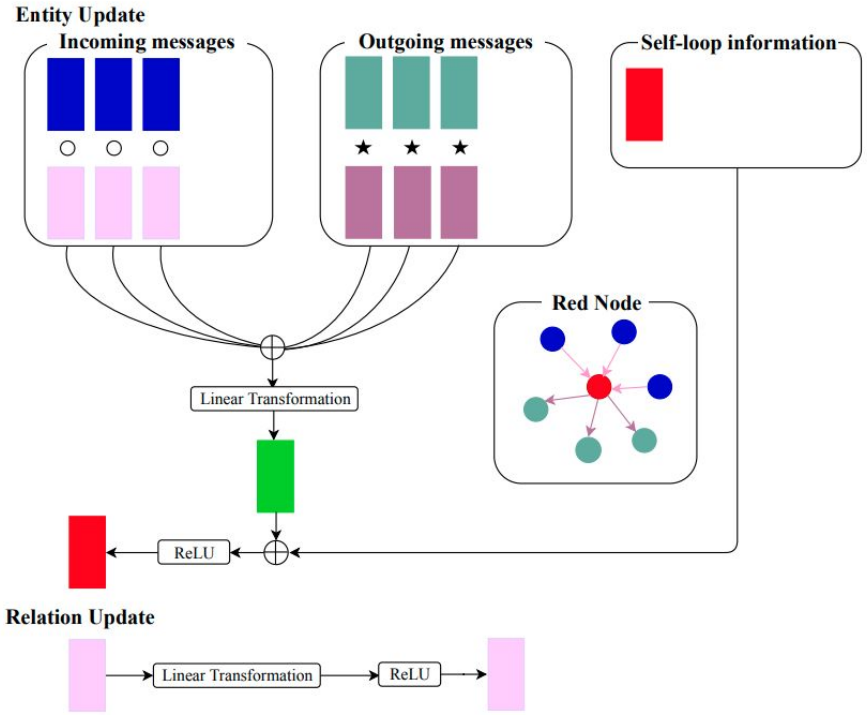
2. Embedding update:

$$h_i^{(l+1)} = \sigma(m_i^{(l+1)} + h_i^{(l)})$$

$$g_k^{(l+1)} = \sigma(W_1^{(l)} g_k^{(l)})$$

TransGCN:Coupling Transformation Assumptions with Graph Convolutional Networks for Link Prediction

GNN-based KG inference: GCN



GNN-based KG inference: GCN

$$m_i^{(l+1)} = \frac{1}{c_i} W_0^{(l)} \left(\sum_{(e_j, r_k, e_i) \in T_{in}(e_i)} (h_j^{(l)} \circ g_k^{(l)}) + \sum_{(e_i, r_k, e_j) \in T_{out}(e_i)} (h_j^{(l)} \star g_k^{(l)}) \right)$$

We want the “dot” and “star” operators to translate the entities to $h_i^{(l)}$.
Easiest assumption, TransE inspired - “dot” is plus, “star” is minus:

$$h_i^{(l)} = \begin{cases} h_j^{(l)} + g_k^{(l)}, (e_j, r_k, e_i) \in T_{in}(e_i) \\ h_j^{(l)} - g_k^{(l)}, (e_i, r_k, e_j) \in T_{out}(e_i) \end{cases}$$

Loss function is also analogous to TransE:

$$L = \sum_{(e_i, r_k, e_j) \in S} \sum_{(e_i', r_k, e_j') \in S'} [\gamma + f_{g_k}(h_i, h_j) - f_{g_k}(h_i', h_j')]_+$$

GNN-based KG inference: GAT

Graph attention-based embedding in KG (GATE-KG):

$$\mathbf{c}_{ijk}^{(l)} = W_1^{(l)}[\mathbf{h}_i^{(l)} || \mathbf{h}_j^{(l)} || \mathbf{g}_k^{(l)}]$$

$$\beta_{ijk}^{(l)} = \text{LeakyReLU}\left(W_2^{(l)}\mathbf{c}_{ijk}^{(l)}\right)$$

$$\alpha_{ijk}^{(l)} = \frac{\exp(\beta_{ijk}^{(l)})}{\sum_{j' \in \mathcal{N}_i}\sum_{k' \in \mathcal{R}_{ij'}} \exp(\beta_{ij'k'}^{(l)})}$$

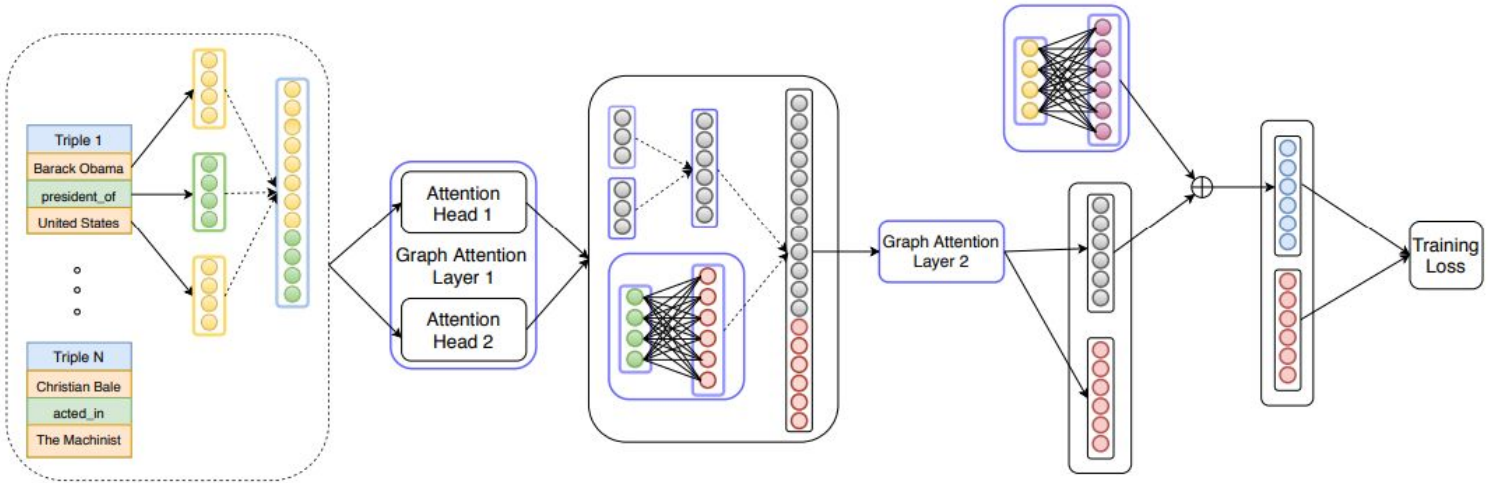
$$\mathbf{h}_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}_i}\sum_{k \in \mathcal{R}_{ij}} \alpha_{ijk}^{(l)}\mathbf{c}_{ijk}^{(l)}\right)$$

Loss function is also analogous to TransE.
For edge prediction, ConvKB is used as a decoder.

Learning Attention-based Embeddings for Relation Prediction in Knowledge Graphs

GNN-based KG inference: GAT

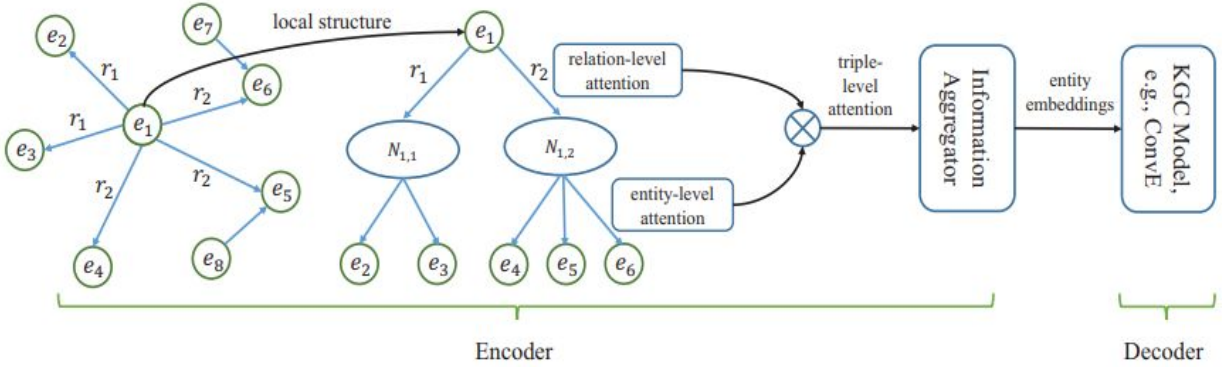
Graph attention-based embedding in KG (GATE-KG):



Learning Attention-based Embeddings for Relation Prediction in Knowledge Graphs

GNN-based KG inference: GAT

Relational Graph neural network with Hierarchical ATtention (RGHAT)



$$\mathbf{a}_{ik} = W_1 [\mathbf{h}_i || \mathbf{g}_k]$$

$$\alpha_{ik} = \frac{\exp(\sigma(\mathbf{z}_1 \cdot \mathbf{a}_{ik}))}{\sum_{r_x \in \mathcal{N}_i} \exp(\sigma(\mathbf{z}_1 \cdot \mathbf{a}_{ix}))}$$

$$\mathbf{b}_{ikj} = W_2 [\mathbf{a}_{ik} || \mathbf{h}_j]$$

$$\beta_{kj} = \frac{\exp(\sigma(\mathbf{z}_2 \cdot \mathbf{b}_{ikj}))}{\sum_{r_y \in \mathcal{N}_{i,k}} \exp(\sigma(\mathbf{z}_1 \cdot \mathbf{b}_{iyj}))}$$

$$\mu_{ikj} = \alpha_{ik} \cdot \beta_{kj}.$$

$$\hat{\mathbf{h}} = \sum_{r \in \mathcal{N}_h} \sum_{t \in \mathcal{N}_{h,r}} \mu_{h,r,t} \mathbf{b}_{h,r,t}$$

[Relational Graph Neural Network with Hierarchical Attention for Knowledge Graph Completion]

Real-life applications

Drug repurposing

Drug repurposing process:

1. **Hypothesis generation**
2. Assessment
3. Validation

KG can be helpful in the hypothesis generation step and has been used to investigate potential drugs for COVID-19:

- Repurpose Open Data to Discover Therapeutics for COVID-19 Using Deep Learning: used RotatE (modification of TransE); identified 41 potential drugs, from which 9 were under clinical trials
- Drug Repurposing for COVID-19 using Graph Neural Network with Genetic, Mechanistic, and Epidemiological Validation: used variational graph autoencoder and transfer learning, identified 22 potential drugs

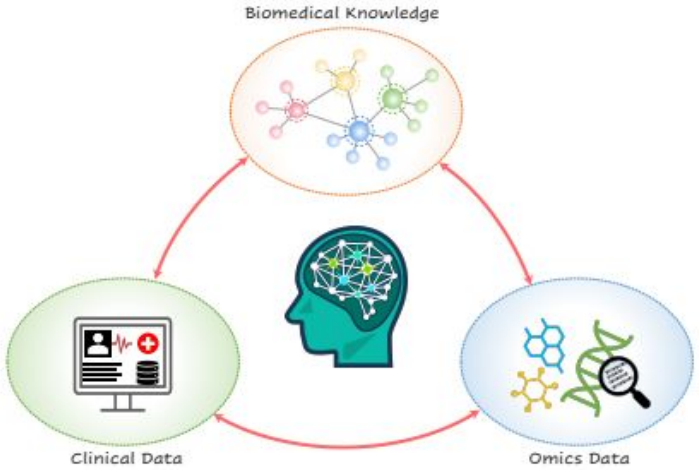
Limitations & future directions

Limitations:

- Data quality:
  - incorrectness
  - incompleteness
- Scalability:
  - KG can include hundreds of millions of relations
  - it can be a challenge for complex GNNs

Future directions:

- Data quality control
- Improving scalability
- Incorporating other data sources:
  - To improve robustness against data quality problems



Thanks for your attention!