Introduction to Graph Representation Learning

Galadriel Brière - marie-galadriel.briere@univ-amu.fr

Notes

- Before the hands-on: **set up your conda environment!!!**

```
conda create --name torch_pyg python=3.10
conda activate torch_pyg
pip install torch
pip install torchkge
pip install pandas matplotlib numpy pyyaml tqdm
pip install pytorch-ignite
pip install ipykernel
```

- **Make sure you can run a Jupyter Notebook with your new conda environment!**
- The **reports from the hands-on will be evaluated + final exam**

Introduction and Key Concepts

Key Concept: **Systems Biology**

- Computational and mathematical analysis and modeling of **complex biological systems**.

  - **Complex Systems**

Systems composed of many components
These components may interact with each others
Properties emerge from these interactions

The whole is greater than the sum of its parts

Key Concept: **Systems Biology**

➢ Computational and mathematical analysis and modeling of **complex biological systems**.

- **Complex Systems**

Systems composed of many components
These components may interact with each others
Properties emerge from these interactions

The whole is greater than the sum of its parts

(a) (b)

Source: CatalyzeX. “DNN-Buddies: A Deep Neural Network-Based Estimation Metric for the Jigsaw Puzzle Problem: Paper and Code.” CatalyzeX. Accessed September 1, 2023. https://www.catalyzex.com/.

Key Concept: **Systems Biology**

- Computational and mathematical analysis and modeling of **complex biological systems**.

Complex Systems

Systems composed of many components
These components may interact with each others
Properties emerge from these interactions

The whole is greater than the sum of its parts

Systems Biology

Systems components: genes/proteins, …
Interactions: PPI, co-expression, …
Emerging properties: Phenotypes

Phenotype does not emerge from isolated biological molecules but from their interactions

Ecological Systems

Systems components: organisms, …
Interactions: prey, symbiosis, competition, …
Emerging properties: resilience, stability, …

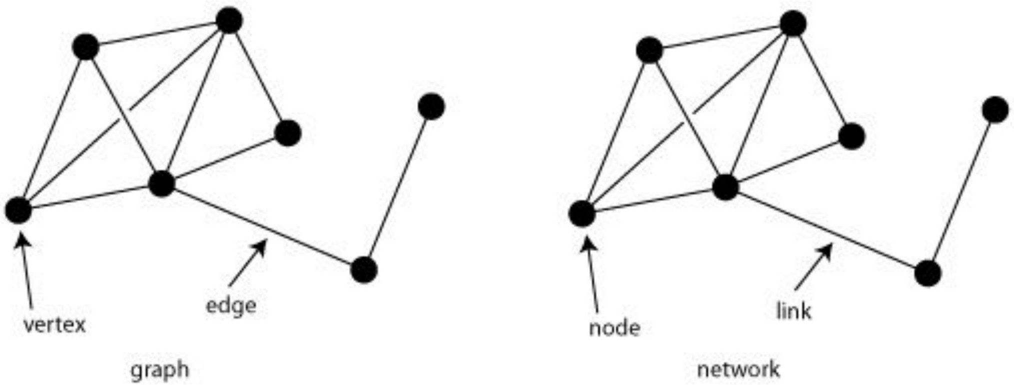
Nervous System

Systems components: neurons, axons, dendrites, …
Interactions: synaptic transmission, …
Emerging properties: memory, cognition, …

Key Concept: **Networks, Graphs and Knowledge Graphs**

- Networks are real-world systems modeled using graphs.

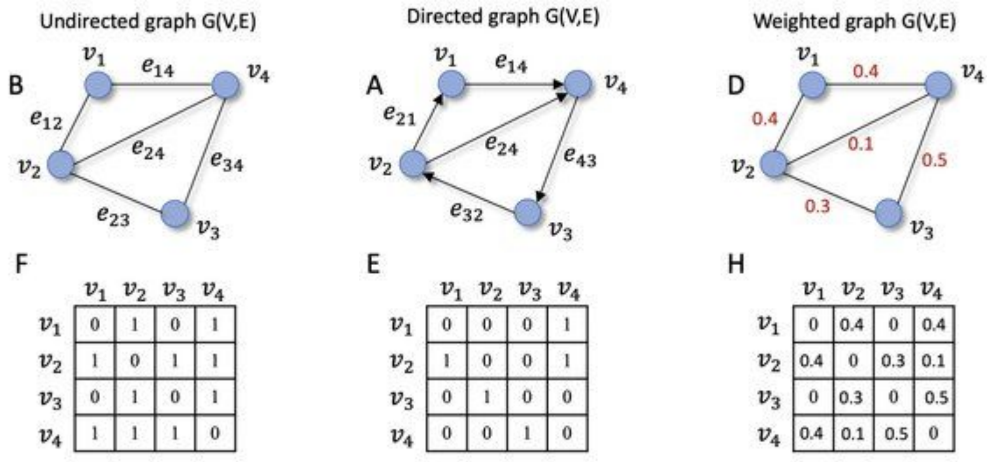
$$G = (V, E), \text{ where } V \text{ is the set of vertices and } E \subseteq (V*V) \text{ is the set of edges}$$



- In practice, the terms network and graph are often used interchangeably.

Key Concept: **Networks, Graphs and Knowledge Graphs**

- Graphs are defined by an Adjacency Matrix
- They can be **undirected**, **directed** and/or **weighted**

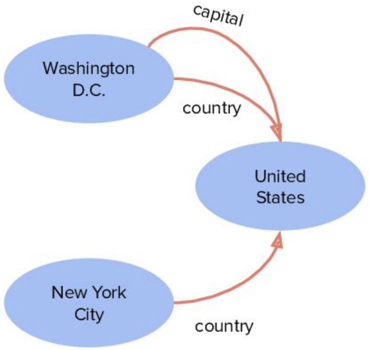


Undirected -> Symmetric adjacency

Directed -> Asymmetric adjacency

Key Concept: **Networks, Graphs and Knowledge Graphs**

➢ **Knowledge Graphs** are structured representations of knowledge in the form or graphs with **semantically enriched edges** (i.e. relations are typed)

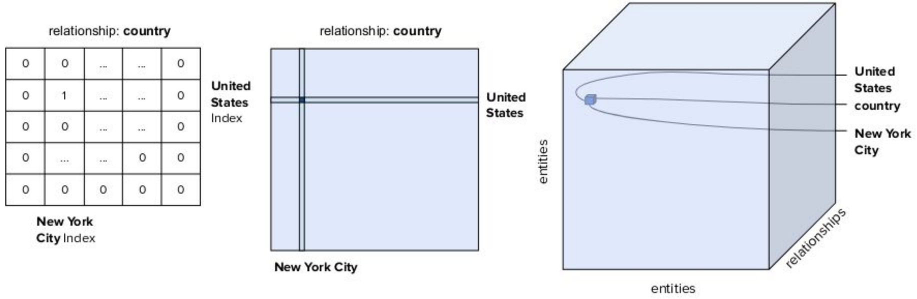


Bianchi, Federico, Gaetano Rossiello, Luca Costabello, Matteo Palmonari, and Pasquale Minervini. "Knowledge Graph Embeddings and Explainable AI," April 30, 2020. https://doi.org/10.3233/SSW200011.

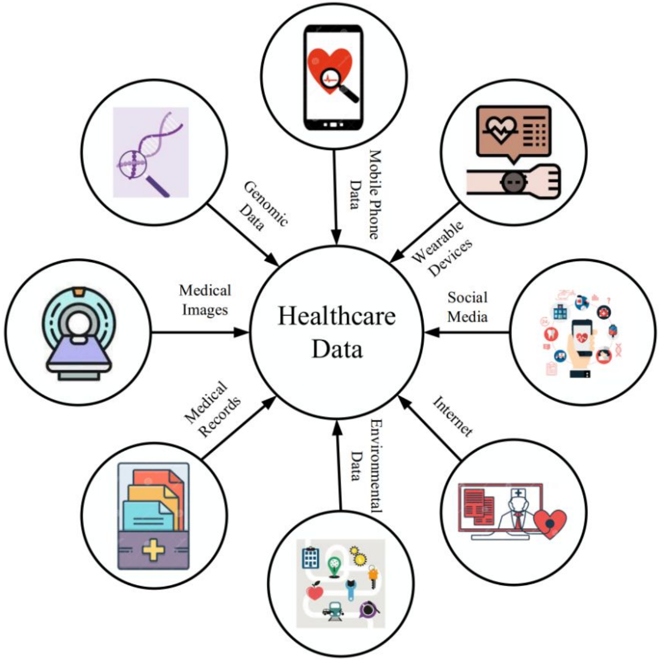
A KG is usually defined as a **set of triplets**:

(head, relation, tail)
or *(h, r, t)*

or a **Tensor of adjacency matrices**:

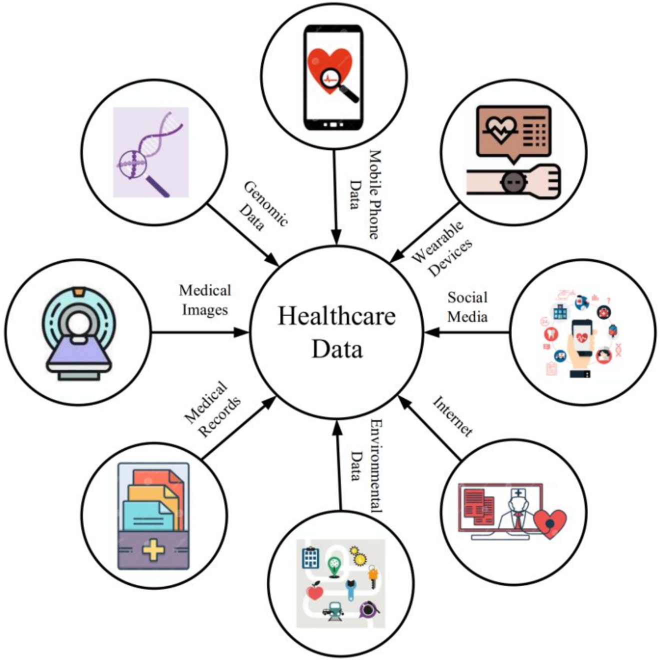


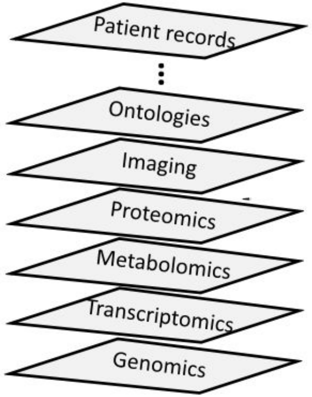
Knowledge Graphs in Biomedicine



Rahmani, Amir Masoud, Efat Yousefpoor, Mohammad Sadegh Yousefpoor, Zahid Mehmood, Amir Haider, Mehdi Hosseinzadeh, and Rizwan Ali Naqvi. 2021. "Machine Learning (ML) in Medicine: Review, Applications, and Challenges" *Mathematics* 9, no. 22: 2970. https://doi.org/10.3390/math9222970

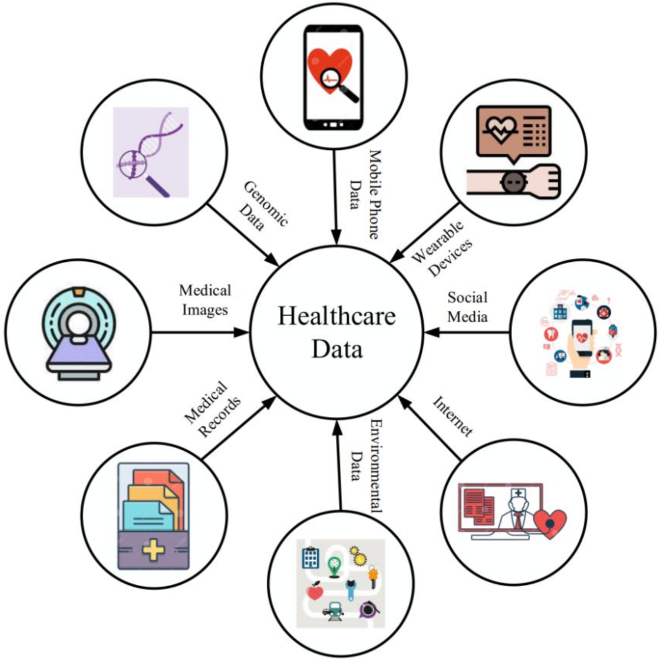
Knowledge Graphs in Biomedicine



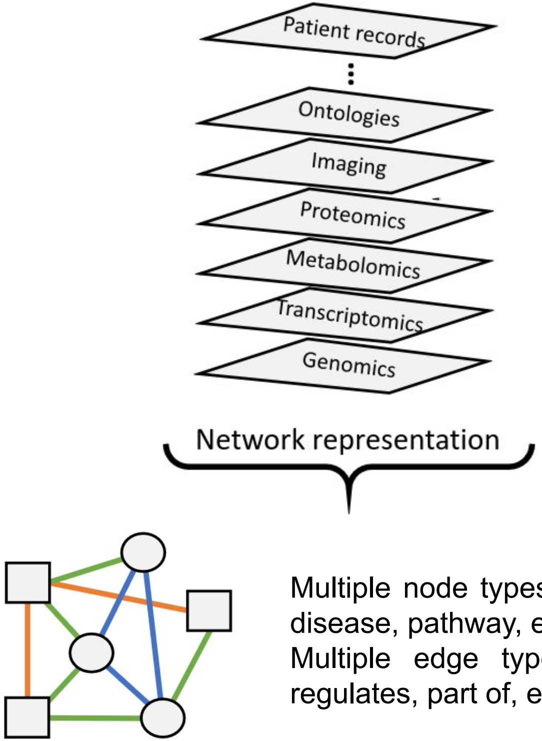


Rahmani, Amir Masoud, Efat Yousefpoor, Mohammad Sadegh Yousefpoor, Zahid Mehmood, Amir Haider, Mehdi Hosseinzadeh, and Rizwan Ali Naqvi. 2021. "Machine Learning (ML) in Medicine: Review, Applications, and Challenges" *Mathematics* 9, no. 22: 2970. https://doi.org/10.3390/math9222970

Knowledge Graphs in Biomedicine



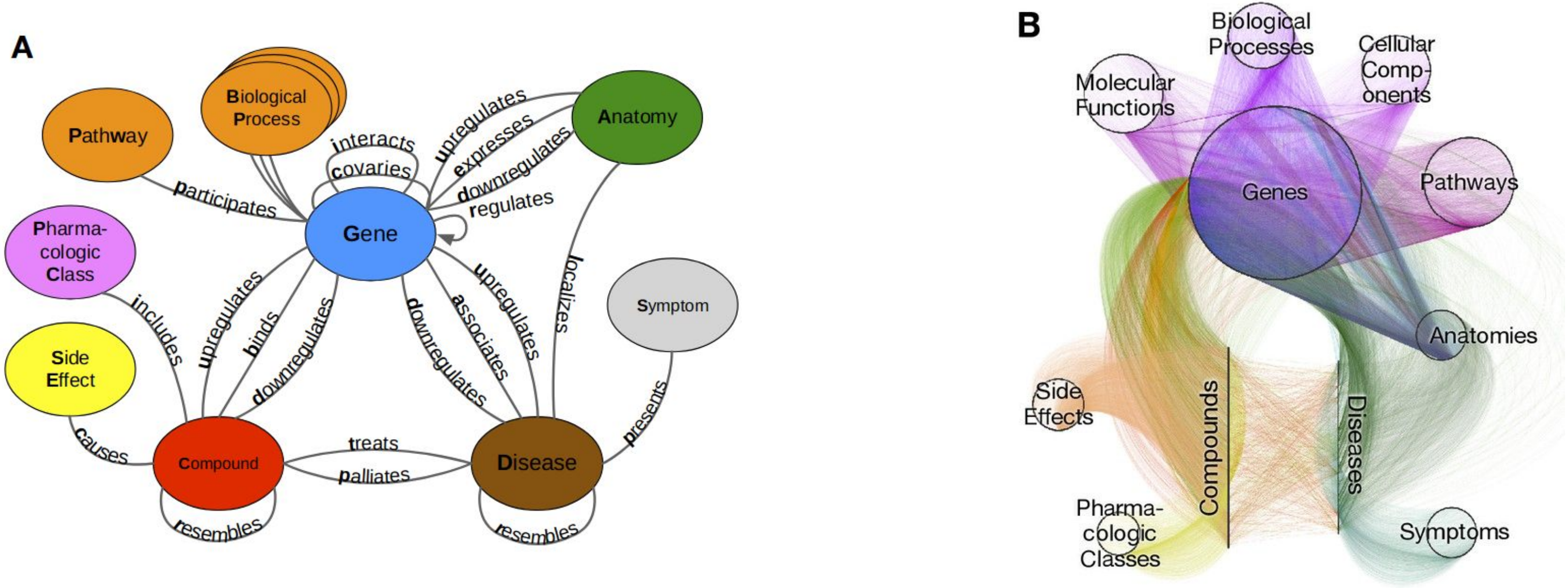
Rahmani, Amir Masoud, Efat Yousefpoor, Mohammad Sadegh Yousefpoor, Zahid Mehmood, Amir Haider, Mehdi Hosseinzadeh, and Rizwan Ali Naqvi. 2021. "Machine Learning (ML) in Medicine: Review, Applications, and Challenges" *Mathematics* 9, no. 22: 2970. https://doi.org/10.3390/math9222970



Multiple node types (e.g. protein, disease, pathway, etc.)
Multiple edge types (e.g. PPI, regulates, part of, etc.)

Zitnik, Marinka, Michelle M. Li, Aydin Wells, Kimberly Glass, Deisy Morselli Gysi, Arjun Krishnan, T. M. Murali, et al. "Current and Future Directions in Network Biology." arXiv, September 15, 2023. https://doi.org/10.48550/arXiv.2309.08478.

Knowledge Graphs in Biomedicine



Himmelstein, Daniel Scott, Antoine Lizee, Christine Hessler, Leo Brueggeman, Sabrina L Chen, Dexter Hadley, Ari Green, Pouya Khankhanian, and Sergio E Baranzini. “Systematic Integration of Biomedical Knowledge Prioritizes Drugs for Repurposing.” eLife 6 (September 22, 2017): e26726. https://doi.org/10.7554/eLife.26726.

Biomedical KGs: statistics

Node Type	Count	Percent (%)	Data Sources
Biological process	28,642	22.1	CTD, Entrez Gene, Gene Ontology
Protein	27,671	21.4	Bgee, CTD, DisGeNET, DrugBank, Entrez Gene, Human Phenotype Ontology, Human PPI Network, Reactome, UMLS
Disease	17,080	13.2	CTD, DisGeNET, Disease Ontology, Drug Central, Human Phenotype Ontology, Mayo Clinic, MONDO Disease Ontology, Orphanet
Phenotype	15,311	11.8	DisGeNET, Human Phenotype Ontology, SIDER
Anatomy	14,035	10.8	Bgee, UBERON
Molecular function	11,169	8.6	CTD, Entrez Gene, Gene Ontology
Drug	7,957	6.2	DrugBank, Drug Central, SIDER
Cellular component	4,176	3.2	CTD, Entrez Gene, Gene Ontology
Pathway	2,516	1.9	Reactome
Exposure	818	0.6	CTD
Total	129,375	100.0	20 primary data sources

Chandak, P., Huang, K. & Zitnik, M. Building a knowledge graph to enable precision medicine. Sci Data 10, 67 (2023). https://doi.org/10.1038/s41597-023-01960-3

Thousands of entities
Millions of relations
KG can be very **complex**, **large** and **dense**, **making their exploration difficult**

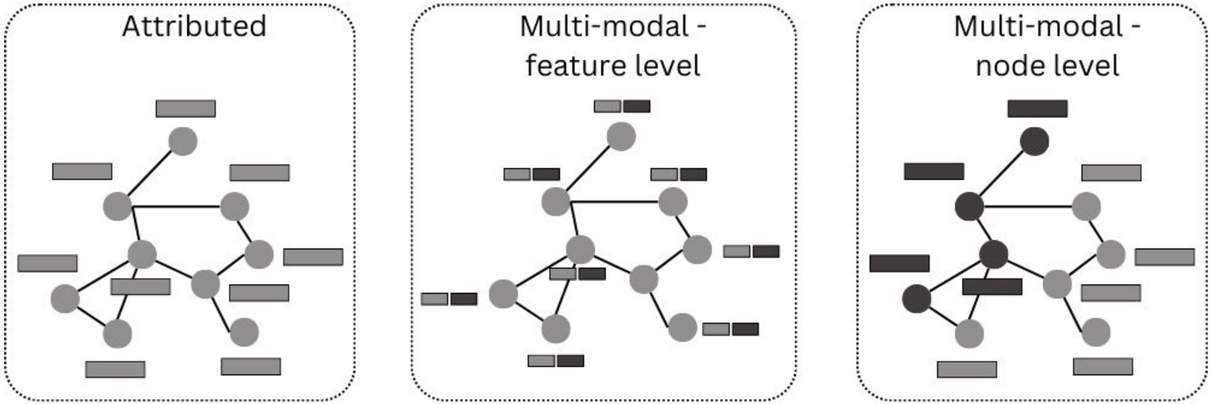
Relation type	Count	Percent (%)
Anatomy - Protein (present)	3,036,406	37.5
Drug - Drug	2,672,628	33.0
Protein - Protein	642,150	7.9
Disease - Phenotype (positive)	300,634	3.7
Biological process - Protein	289,610	3.6
Cellular component - Protein	166,804	2.1
Disease - Protein	160,822	2.0
Molecular function - Protein	139,060	1.7
Drug - Phenotype	129,568	1.6
Biological process - Biological process	105,772	1.3
Pathway - Protein	85,292	1.1
Disease - Disease	64,388	0.8
Drug - Disease (contraindication)	61,350	0.8
Drug - Protein	51,306	0.6
Anatomy - Protein (absent)	39,774	0.5
Phenotype - Phenotype	37,472	0.5
Anatomy - Anatomy	28,064	0.3
Molecular function - Molecular function	27,148	0.3
Drug - Disease (indication)	18,776	0.2
Cellular component - Cellular component	9,690	0.1
Phenotype - Protein	6,660	0.1
Drug - Disease (off-label use)	5,136	0.1
Pathway - Pathway	5,070	0.1
Exposure - Disease	4,608	0.1
Exposure - Exposure	4,140	0.1
Exposure - Biological process	3,250	<0.1
Exposure - Protein	2,424	<0.1
Disease - Phenotype (negative)	2,386	<0.1
Exposure - Molecular function	90	<0.1
Exposure - Cellular component	20	<0.1
Total	8,100,498	100.0

Key concept: Multimodal Knowledge Graphs

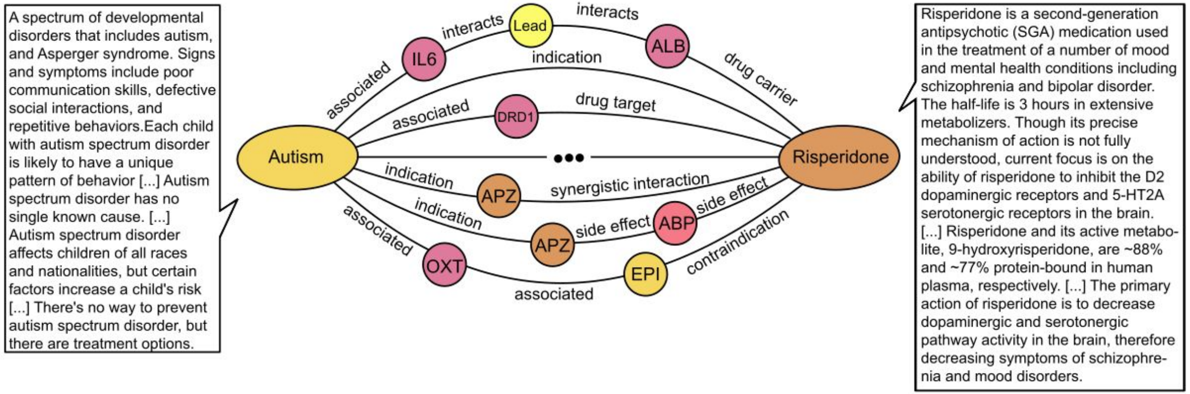
Nodes can be **described with multimodal attributes**:

- ➢ Descriptive text
- ➢ Omics data
- ➢ Images
- ➢ Sequences
- ➢ …

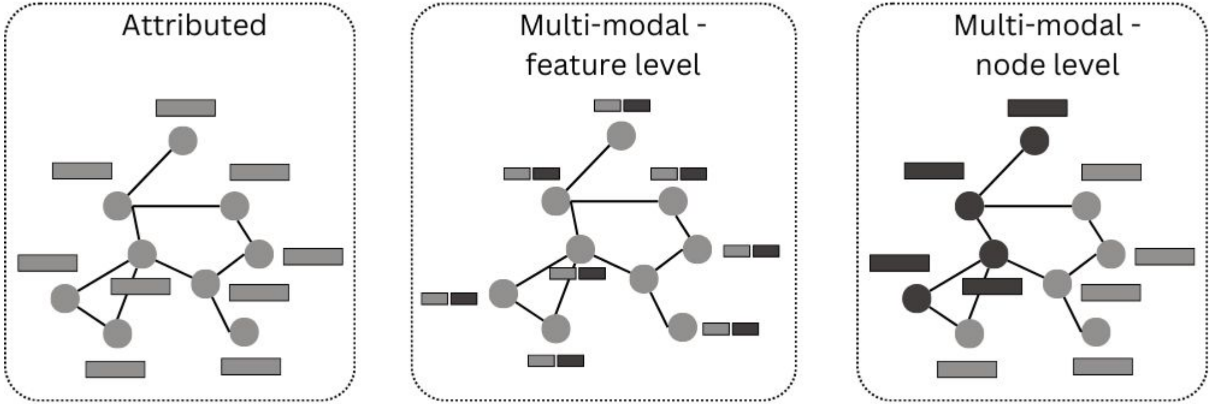
This multi-modal data can be included in the graph structure:



Key concept: Multimodal Knowledge Graphs



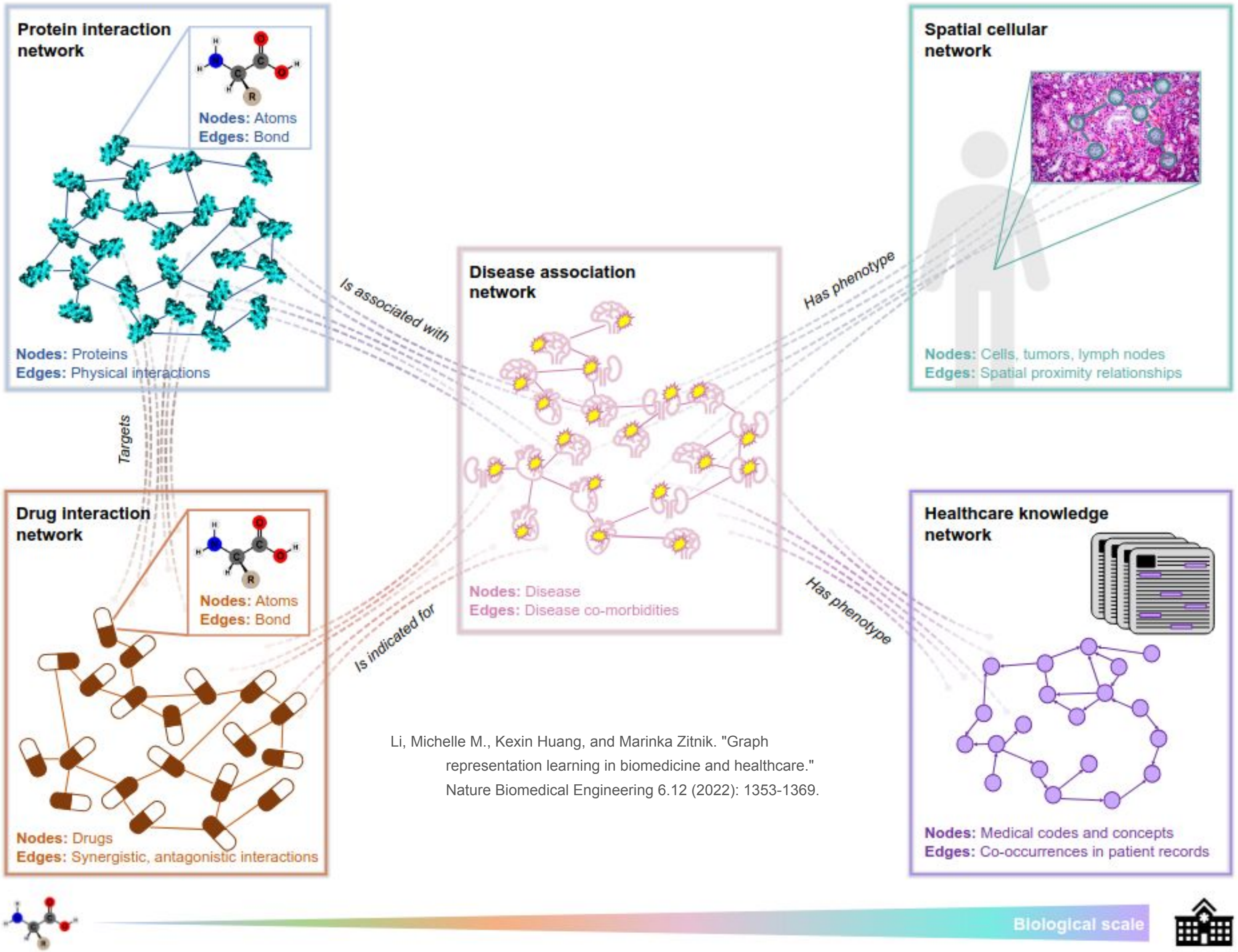
Chandak, P., Huang, K. & Zitnik, M. Building a knowledge graph to enable precision medicine. Sci Data 10, 67 (2023). https://doi.org/10.1038/s41597-023-01960-3



Key

Nodes

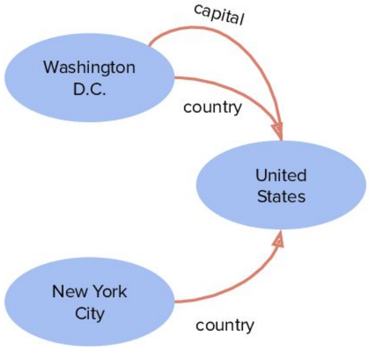
This m

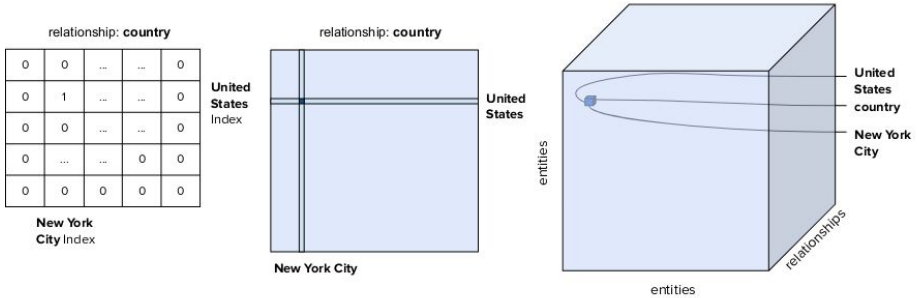


Li, Michelle M., Kexin Huang, and Marinka Zitnik. "Graph representation learning in biomedicine and healthcare." Nature Biomedical Engineering 6.12 (2022): 1353-1369.

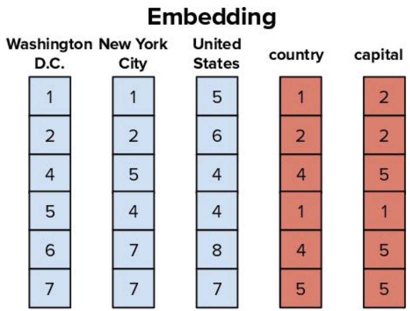
Key Concept 3: **Knowledge Graph Embedding**

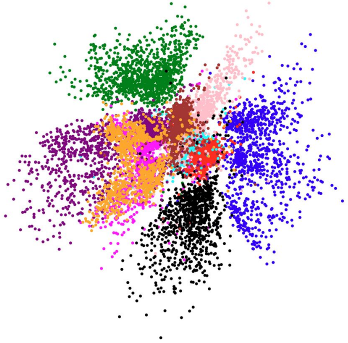
Bianchi, Federico, Gaetano Rossiello, Luca Costabello, Matteo Palmonari, and Pasquale Minervini. “Knowledge Graph Embeddings and Explainable AI,” April 30, 2020. https://doi.org/10.3233/SSW200011.





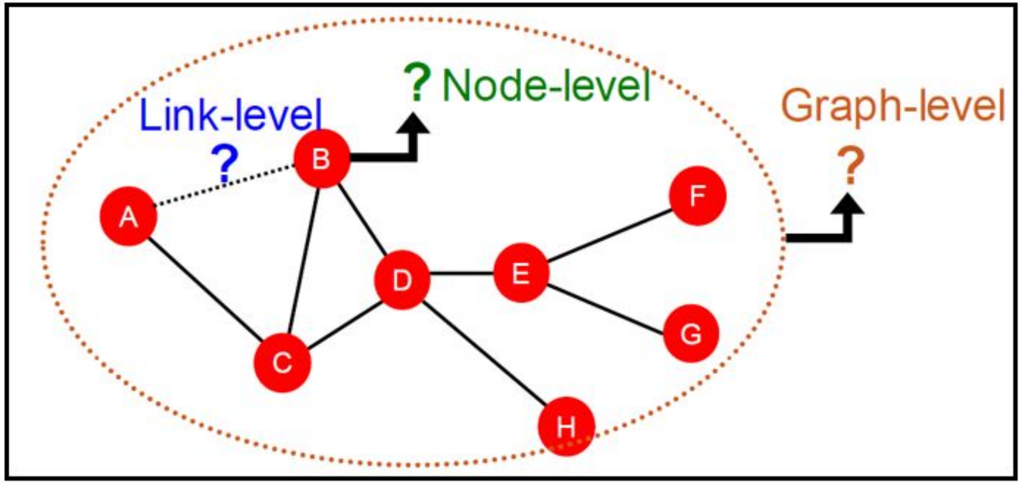
High-dimensional node vector representation





Encoding nodes and relations in a lower-dimensional space

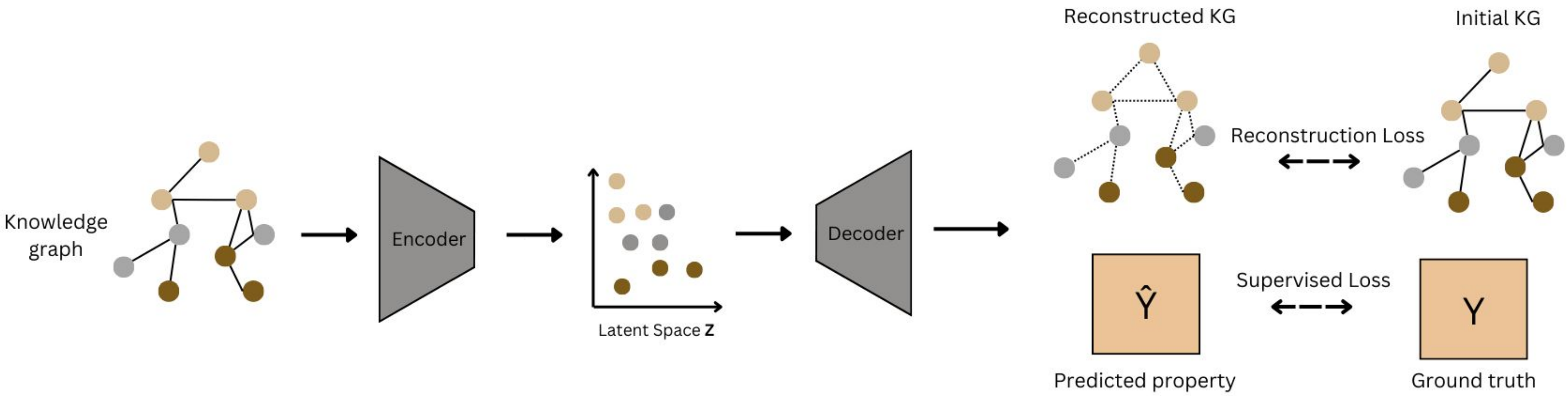
Tasks in Knowledge Graph Embedding



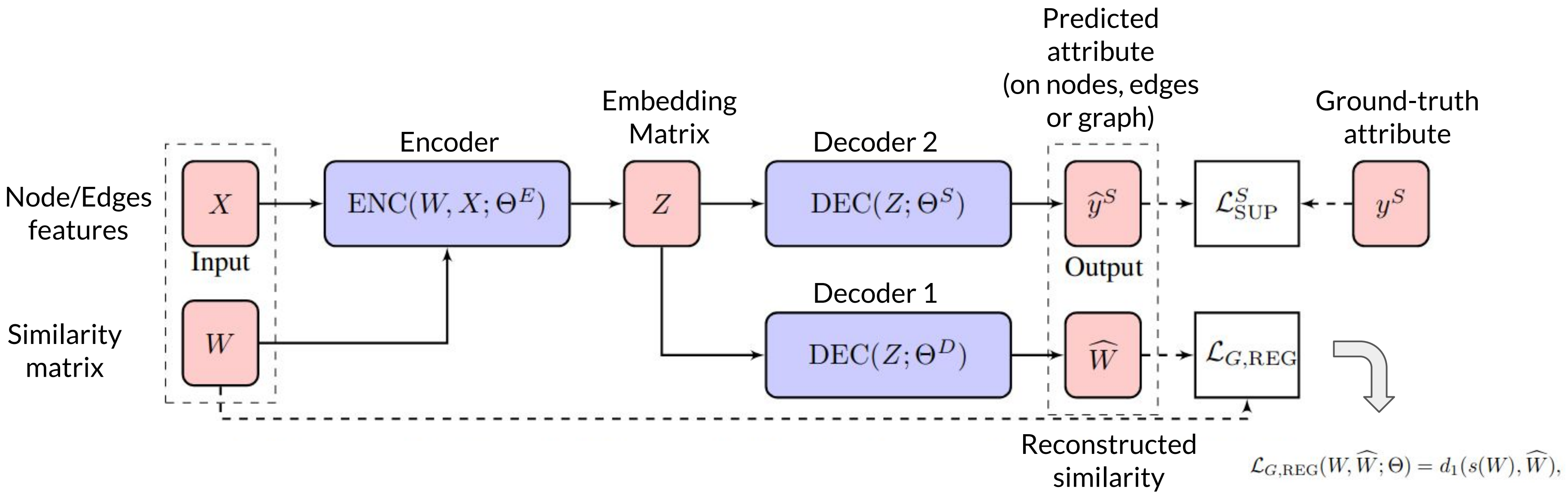
- ➢ **Classification**
  - Posititive/negative triplet
  - Node properties
  - Graph properties
- ➢ **Clustering**
  - Node clustering
  - Graph clustering
  - (Edge clustering)
- ➢ **Regression**
  - Node properties
  - Edge weights
  - Graph properties

Ajay Taneja, Extracting Node Level Features from Graph Networks for Machine Learning Models, LinkedIn post

Key Concept: **KGE as Auto-Encoder Frameworks**

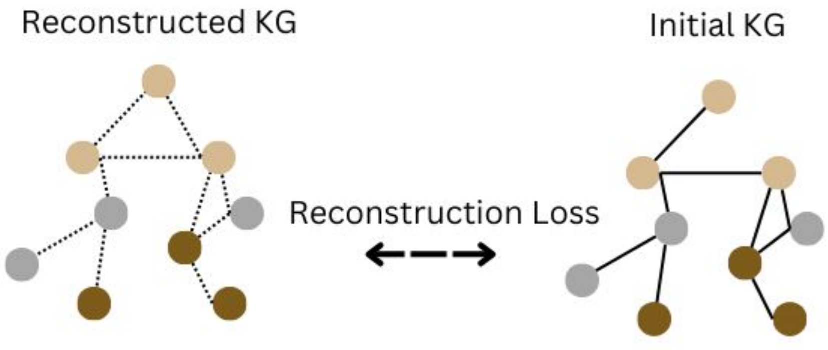


Key Concept: **KGE as Auto-Encoder Frameworks**



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

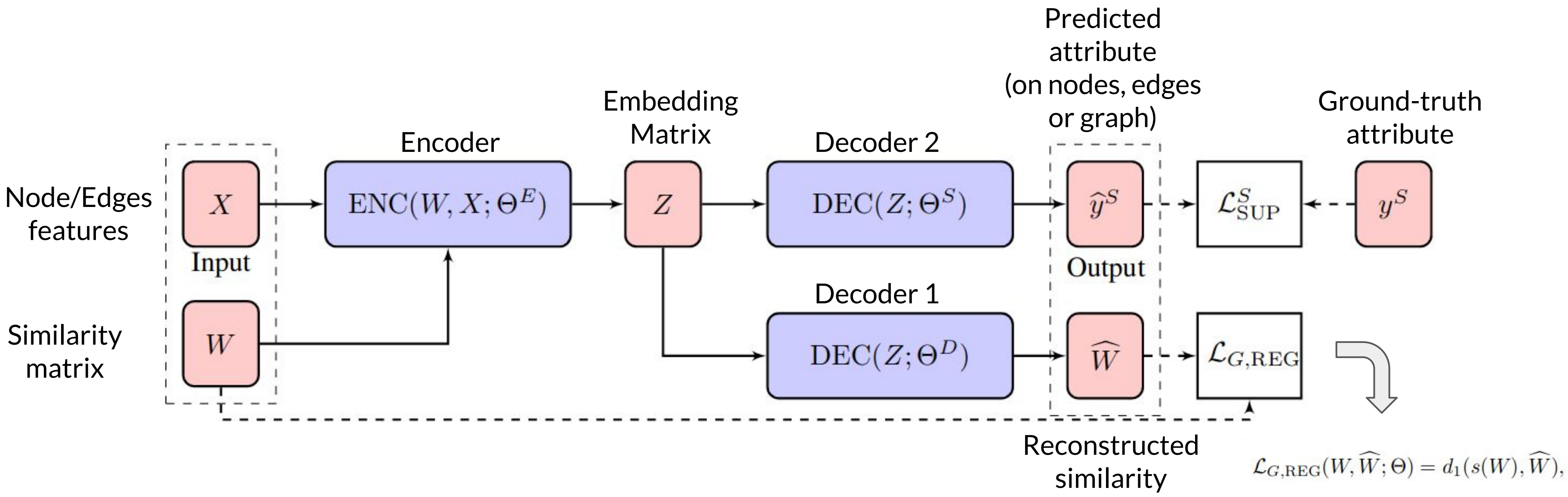
Key concept: **Negative Sampling**



In practice, these models are trained to embed **entities from true triplets closer to each other than those from negative triplets**. These negative triplets are generated by randomly replacing either the head or tail entity for a given triplet: this is called **Negative Sampling.**

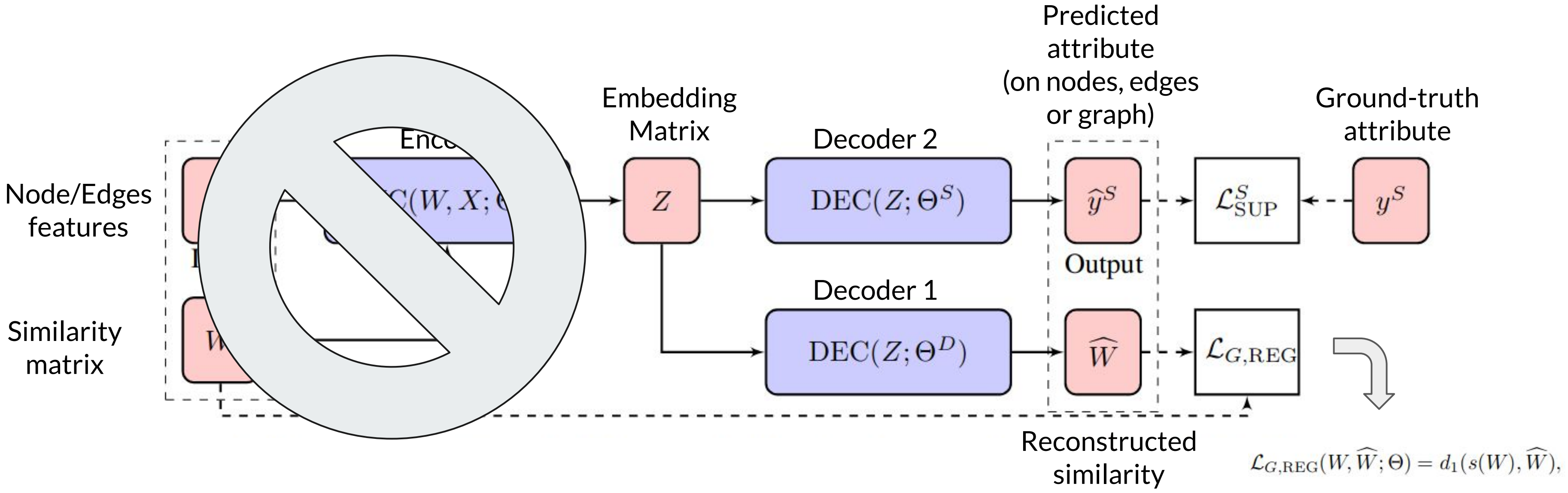
Overview of KG embedding strategies

Shallow VS Deep Embedding



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

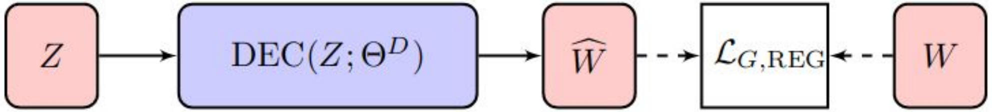
Shallow VS Deep Embedding



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

Shallow embedding

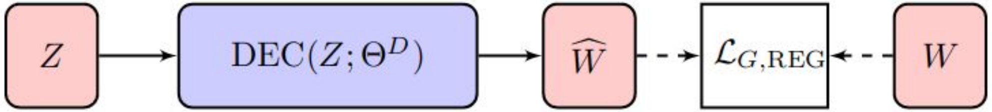
The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

Shallow embedding

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

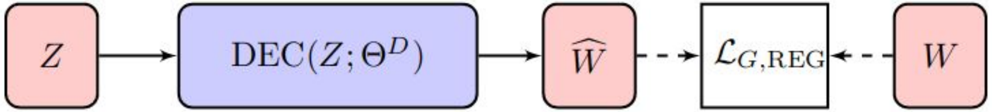
Translational Models

Bilinear Models

Random Walk based Models

Shallow embedding

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

Translational Models

Bilinear Models

Random Walk based Models

Shallow Embedding: Translational Models

$$\text{Score}(\mathbf{h}, \mathbf{r}, \mathbf{t}) = -||\phi(\mathbf{h}, \mathbf{r}) - \psi(\mathbf{t}, \mathbf{r})||_p$$

Where:

- $\phi(\mathbf{h}, \mathbf{r})$: Transformation of the head entity embedding $\mathbf{h}$ based on the relation embedding $\mathbf{r}$.
- $\psi(\mathbf{t}, \mathbf{r})$: Transformation of the tail entity embedding $\mathbf{t}$ based on the relation embedding $\mathbf{r}$.
- $|| \cdot ||_p$: Norm (e.g., $p = 1$ for L_1-norm or $p = 2$ for L_2-norm).

Shallow Embedding: Translational Models

Margin-based loss

$$\text{Score}(\mathbf{h}, \mathbf{r}, \mathbf{t}) = -\|\phi(\mathbf{h}, \mathbf{r}) - \psi(\mathbf{t}, \mathbf{r})\|_p$$

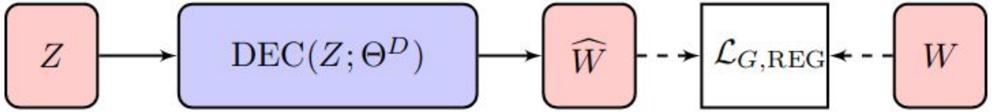
$$\mathcal{L} = \sum_{(h,r,t)\in\mathcal{T}_{\text{train}}} \sum_{(h',r,t')\in\mathcal{T}_{\text{neg}}} \max(0, \gamma + \text{Score}(\mathbf{h}', \mathbf{r}, \mathbf{t}') - \text{Score}(\mathbf{h}, \mathbf{r}, \mathbf{t}))$$

- $\mathcal{T}_{\text{train}}$: Set of positive triples (ground truth).
- $\mathcal{T}_{\text{neg}}$: Set of negative triples generated via corruption (e.g., replacing h or t).
- γ: Margin hyperparameter.
- **Goal**: Separate scores of positive and negative triples by at least γ:

$$\text{Score}(\mathbf{h}, \mathbf{r}, \mathbf{t}) > \text{Score}(\mathbf{h}', \mathbf{r}, \mathbf{t}') + \gamma$$

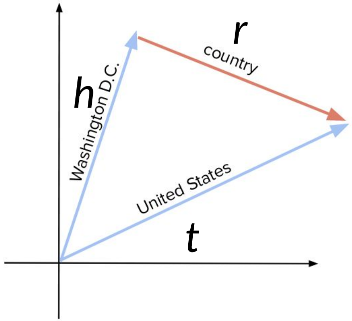
Shallow embedding: **Translational Models**

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

TransE



Bianchi, Federico, Gaetano Rossiello, Luca Costabello, Matteo Palmonari, and Pasquale Minervini. "Knowledge Graph Embeddings and Explainable AI," April 30, 2020. https://doi.org/10.3233/SSW200011.

h + r ≃ t if (h, r, t) holds

$$\mathscr{L} = \sum_{h,r,t \in S} \sum_{h',r,t' \in S'_{h,r,t}} [\gamma + d(\mathbf{h}+\mathbf{r},\mathbf{t}) - d(\mathbf{h}'+\mathbf{r},\mathbf{t}')]_+,$$

Advantages:

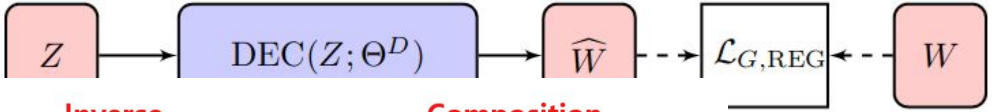
- ➔ Inverse relationships
- ➔ Antisymmetric relationships
- ➔ Compositions

Limitations:

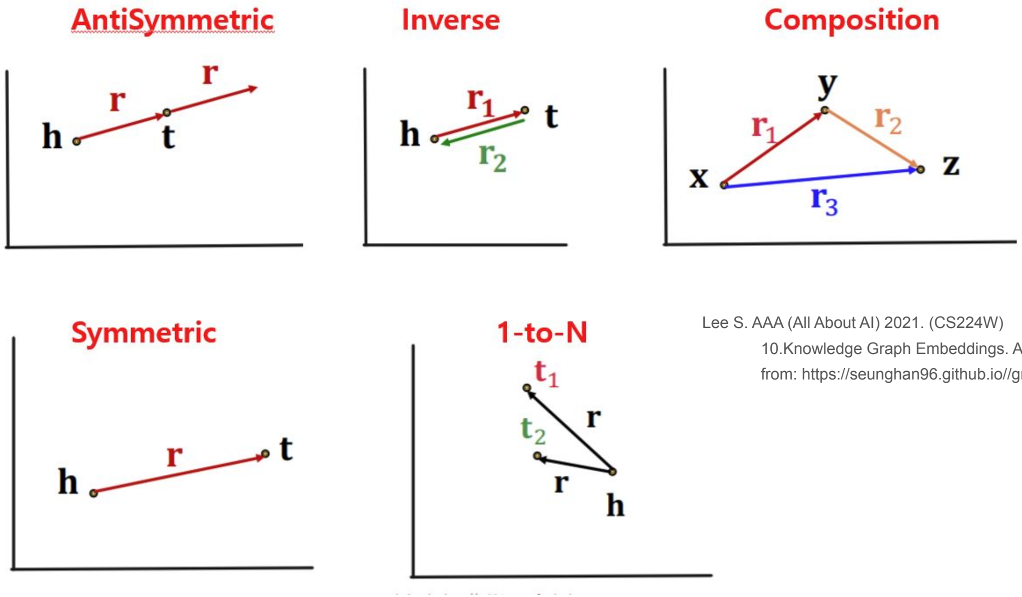
- ➔ One-to-many/many-to-many relationships
- ➔ Symmetric relationships
- ➔ Hierarchical relationships

Shallow embedding: **Translational Models**

The graph structure (W) is only used in the loss function.



:hine Learning on Graphs: A
2005.03675.



Lee S. AAA (All About AI) 2021. (CS224W) 10.Knowledge Graph Embeddings. Available from: https://seunghan96.github.io//gnn/gnn10/

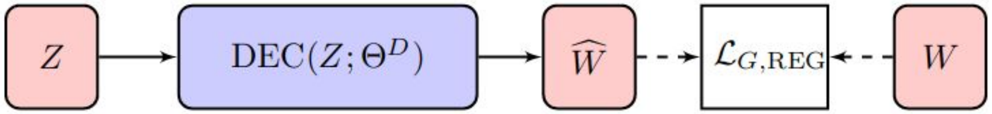
TransE

Advantages:
- ➔ Inverse relationships
- ➔ Antisymmetric relationships
- ➔ Compositions

Limitations:
- ➔ One-to-many/many-to-many relationships
- ➔ Symmetric relationships
- ➔ Hierarchical relationships

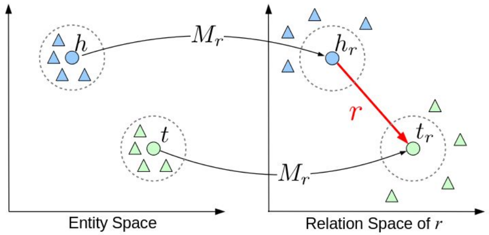
Shallow embedding: **Translational Models**

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

TransR



Lin, Yankai, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. "Learning Entity and Relation Embeddings for Knowledge Graph Completion." Proceedings of the AAAI Conference on Artificial Intelligence 29, no. 1 (February 19, 2015). https://doi.org/10.1609/aaai.v29i1.9491.

$\mathbf{h_r + r \approxeq t_r}$ if (h, r, t) holds

$\mathbf{h_r = h^*M_r}$ with $\mathbf{M_r}$ the projection matrix for relation r

$\mathbf{t_r = t^*M_r}$

Advantages:

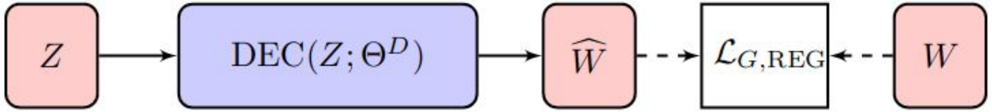
- ➔ Inverse relationships
- ➔ Antisymmetric relationships
- ➔ One-to-many/many-to-many relationships
- ➔ Symmetric relationships

Limitations:

- ➔ Compositions

Shallow embedding

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

Translational Models

Bilinear Models

Random Walk based Models

Shallow Embedding: Bilinear Models

$$\mathrm{Score}(\mathbf{h}, \mathbf{r}, \mathbf{t}) = \mathbf{h}^\top \mathbf{M}_r \mathbf{t}$$

Where:

- $\mathbf{h}$: Embedding of the head entity.
- $\mathbf{t}$: Embedding of the tail entity.
- $\mathbf{M}_r$: Relation-specific transformation matrix that captures how the head and tail interact for the relation $\mathbf{r}$.

Shallow Embedding: Bilinear Models

$$\mathrm{Score}(\mathbf{h}, \mathbf{r}, \mathbf{t}) = \mathbf{h}^{\top} \mathbf{M}_r \mathbf{t}$$

Binary-cross Entropy Loss

$$\mathcal{L} = -\sum_{(h,r,t)} [y \cdot \log(\mathrm{Score}(h, r, t)) + (1 - y) \cdot \log(1 - \mathrm{Score}(h, r, t))]$$

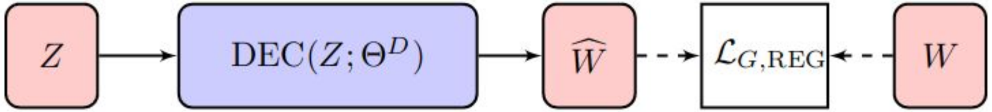
Where:

- $y = 1$ for positive triples, $y = 0$ for negative triples.

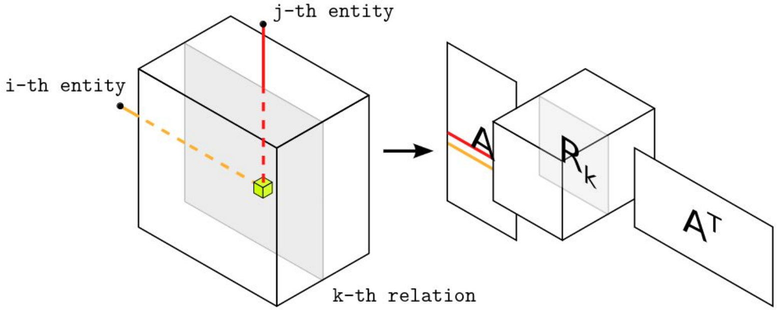
Goal: Maximize the likelihood of assigning higher probabilities to positive triples.

Shallow embedding: **Bilinear Models**

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.



Krompass, Denis, Maximilian Nickel, Xueyan Jiang, and Volker Tresp. Non-Negative Tensor Factorization with RESCAL, 2013. https://doi.org/10.13140/2.1.3725.3124.

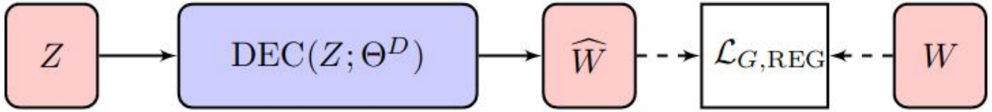
RESCAL

Advantages:
- ➔ Fully expressive
- ➔ Collective learning

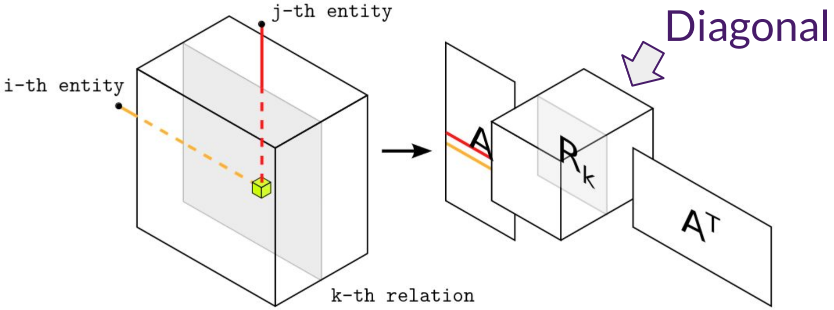
Limitations:
- ➔ Computationally intensive
- ➔ Prone to overfitting

Shallow embedding: **Bilinear Models**

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.



Krompass, Denis, Maximilian Nickel, Xueyan Jiang, and Volker Tresp. Non-Negative Tensor Factorization with RESCAL, 2013. https://doi.org/10.13140/2.1.3725.3124.

DistMult

Advantages:

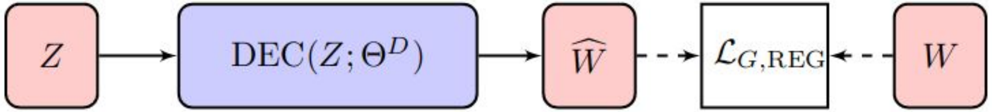
- ➔ Collective learning

Limitations:

- ➔ Cannot differentiate between head and tail (all relations are modeled as symetric)

Shallow embedding

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

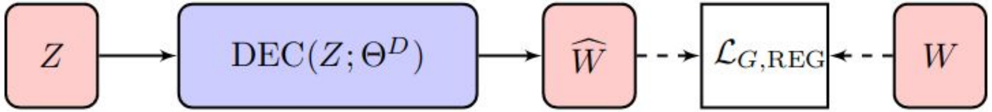
Translational Models

Bilinear Models

Random Walk based Models

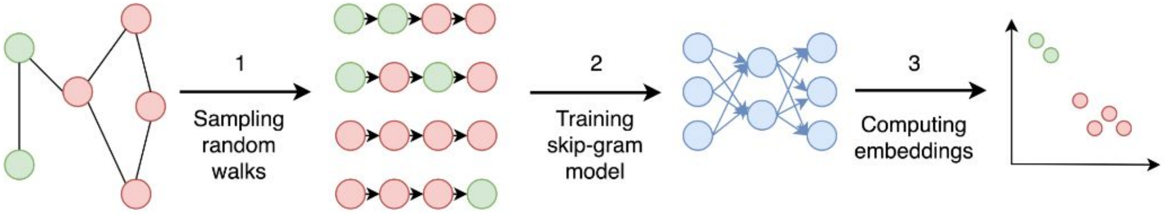
Shallow embedding: **Random Walk-based Models**

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

NLP model: Skip-gram maximizes the co-occurrence probability among the words that appear within a window in a sentence



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

DeepWalk, node2vec, …

Advantages:

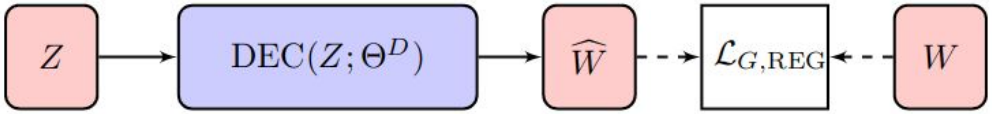
- ➔ Higher-order similarities
- ➔ Capture neighborhood similarities and community membership
- ➔ Many walking strategy

Limitations:

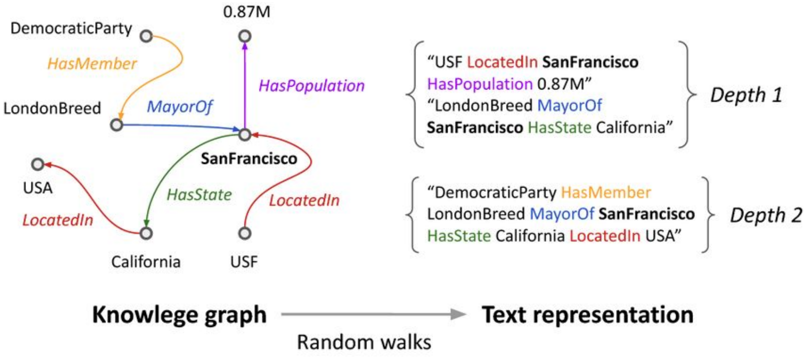
- ➔ For networks (do not consider relationship types)

Shallow embedding: **Random Walk-based Models**

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.



Cvetkov-Iliev, Alexis, Alexandre Allauzen, and Gaël Varoquaux. "Relational Data Embeddings for Feature Enrichment with Background Information." Machine Learning 112, no. 2 (February 2023): 687–720. https://doi.org/10.1007/s10994-022-06277-7.

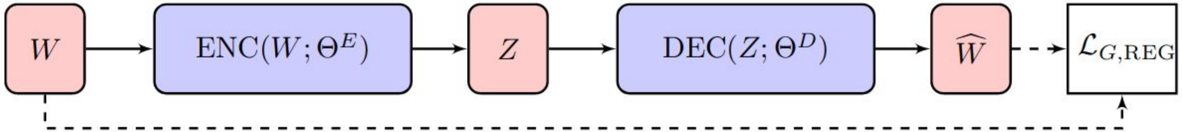
RDF2vec

Advantages:

- ➔ Higher-order similarities
- ➔ Capture neighborhood similarities and community membership

Deep embedding

The graph structure (W) is only used in the encoder and in the loss function.

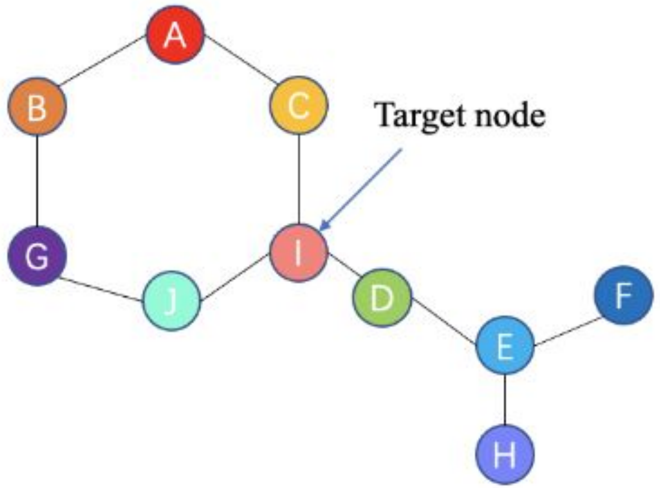


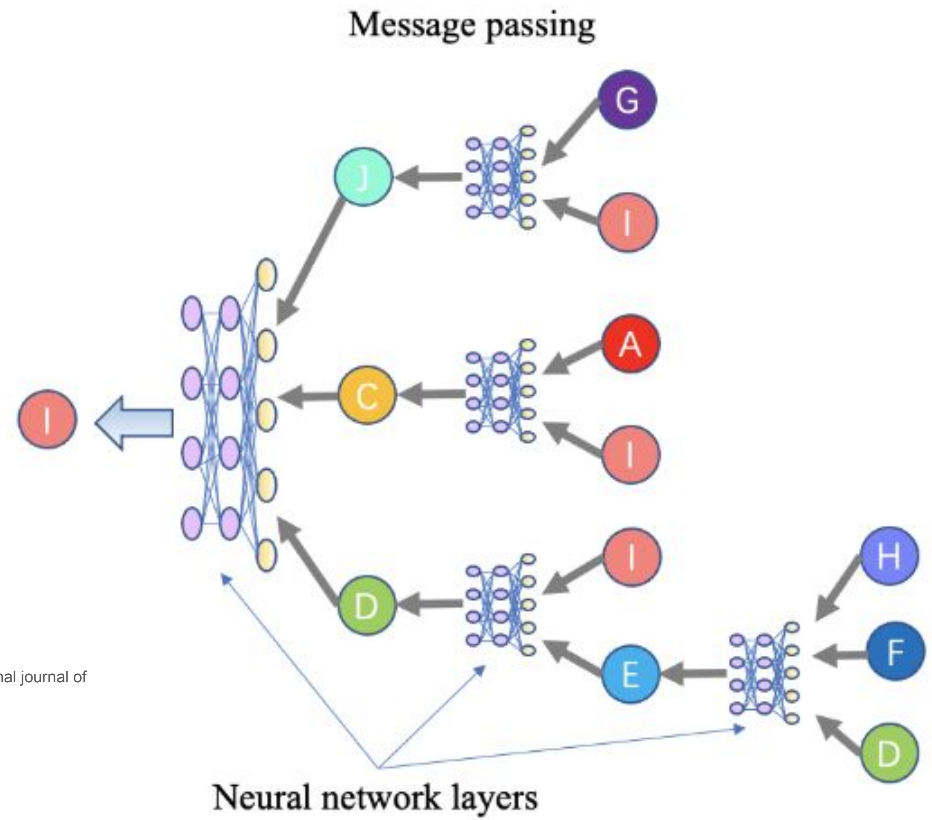
Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

Graph Neural Networks

Graph Transformers

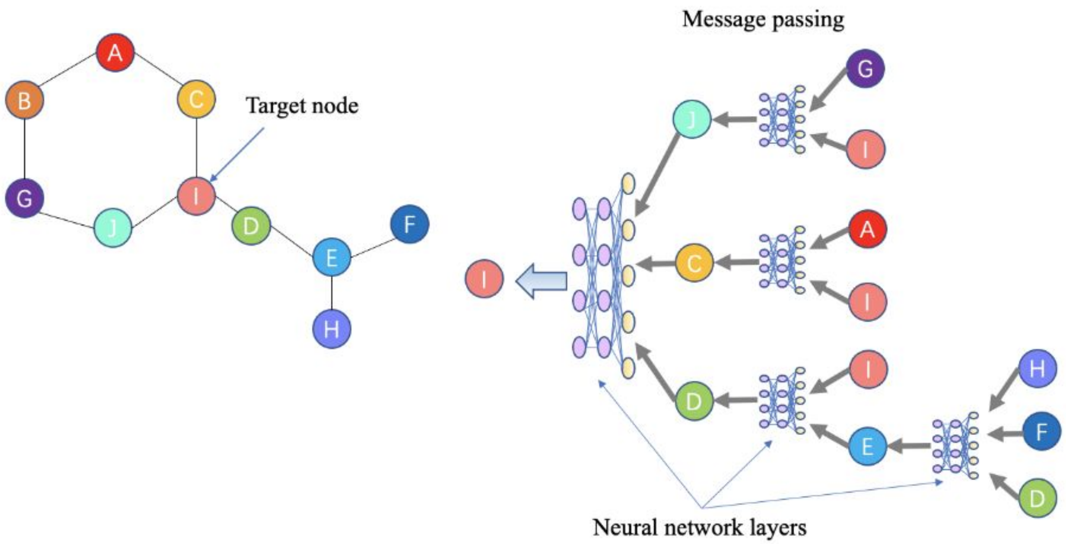
Deep embedding: the concept of **Message Passing**





Zhang, Yue, et al. "Application of computational biology and artificial intelligence in drug design." International journal of molecular sciences 23.21 (2022): 13568.

Deep embedding: the concept of **Message Passing**



Zhang, Yue, et al. "Application of computational biology and artificial intelligence in drug design." International journal of molecular sciences 23.21 (2022): 13568.

Initialisation

- Random
- Shallow embeddings
- Node features

Aggregation

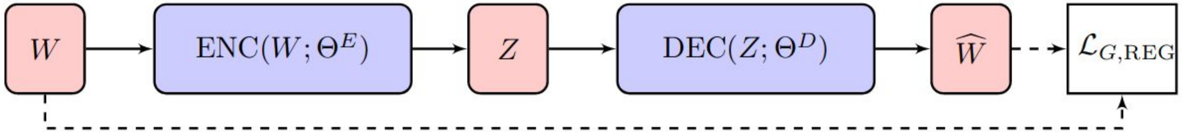
$$m_v^i = A\left(\{s_u^i\}_{u \in N(v)}\right),$$

Update

$$s_v^{i+1} = U\left(s_v^i, m_v^i\right),$$

Deep embedding

The graph structure (W) is only used in the encoder and in the loss function.

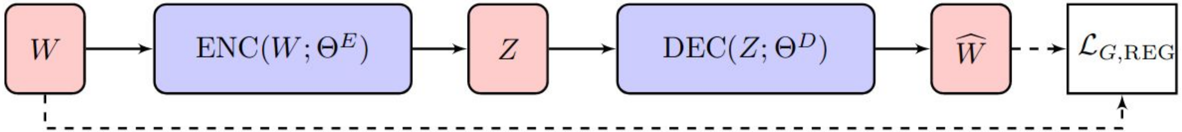


Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. “Machine Learning on Graphs: A Model and Comprehensive Taxonomy.” arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

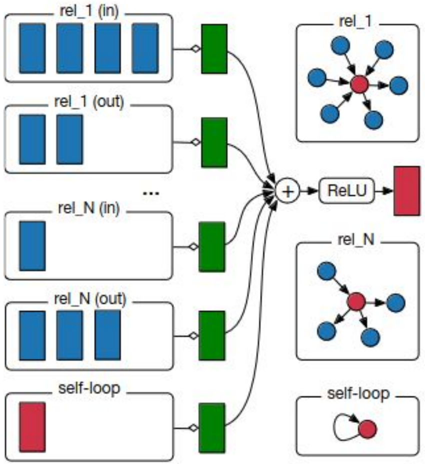
Graph Neural Networks Graph Transformers

Deep embedding: **Graph Neural Networks**

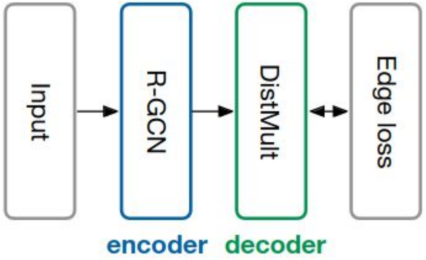
The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.



$$h_i^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)} + W_0^{(l)}h_i^{(l)}\right)$$



Schlichtkrull, Michael, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. "Modeling Relational Data with Graph Convolutional Networks." arXiv, October 26, 2017. http://arxiv.org/abs/1703.06103.

R-GCN

Advantages:

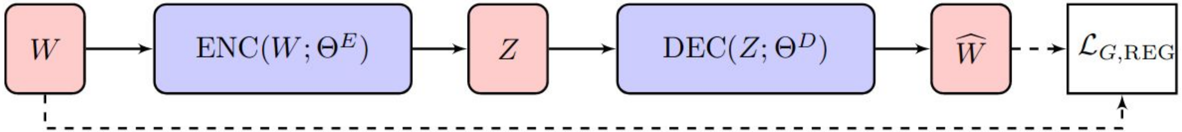
- ➔ Expressive for local and global relationships
- ➔ Inductive

Limitations:

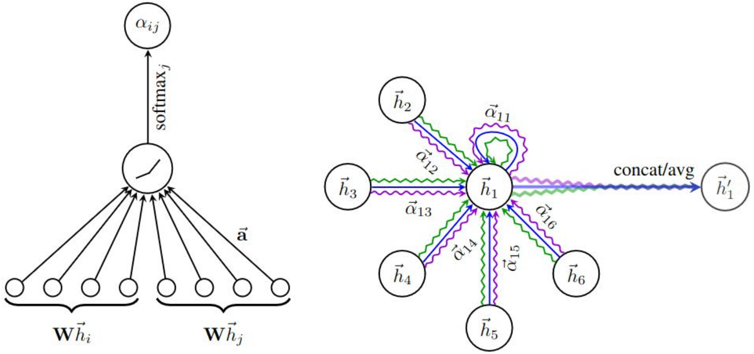
- ➔ All neighbors for relation type r have the same importance when updating a node embedding

Deep embedding: **Graph Attention Networks**

The graph structure (W) is only used in the loss function.



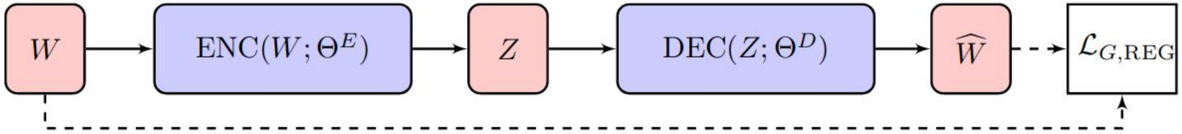
Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. “Machine Learning on Graphs: A Model and Comprehensive Taxonomy.” arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.



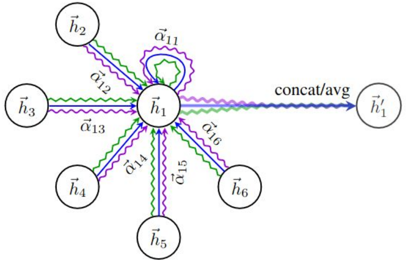
Veličković, Petar, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. “Graph Attention Networks.” arXiv, February 4, 2018. http://arxiv.org/abs/1710.10903.

Deep embedding: **Graph Attention Networks**

The graph structure (W) is only used in the loss function.



Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.



Veličković, Petar, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. "Graph Attention Networks." arXiv, February 4, 2018. http://arxiv.org/abs/1710.10903.

$$\mathrm{att}^k_{viu} = f^k\left[e^k_v || r^k_i || e^k_u\right]$$

Contribution of neighbor u to update node v embedding (for rel i and channel k)

$$e_v^{(l)} = \Big\Vert_{k=1}^{K} \sigma_1\left(\sum_{u\in\mathcal{N}_v}\sum_{i\in\mathcal{R}_{vu}} \alpha^k_{viu}[e^k_u * r^k_i]\right)$$

Update considers all channels K and all neighbors for every relation

Chen, Meiqi, Yuan Zhang, Xiaoyu Kou, Yuntao Li, and Yan Zhang. "R-GAT: Relational Graph Attention Network for Multi-Relational Graphs." arXiv, September 13, 2021. http://arxiv.org/abs/2109.05922.

r-GAT

Advantages:

- ➔ Expressive for local and global relationships
- ➔ Assign a similar embedding to node i and to its more similar or important neighbors
- ➔ Inductive

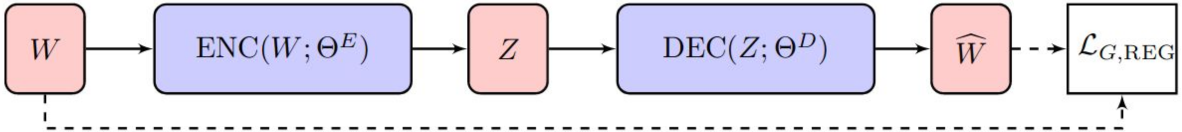
Limitations:

- ➔ Computationally intensive

Brody, Shaked, Uri Alon, and Eran Yahav. "**How Attentive Are Graph Attention Networks?**," 2021. https://openreview.net/forum?id=F72ximsx7C1.

Deep embedding

The graph structure (W) is only used in the encoder and in the loss function.

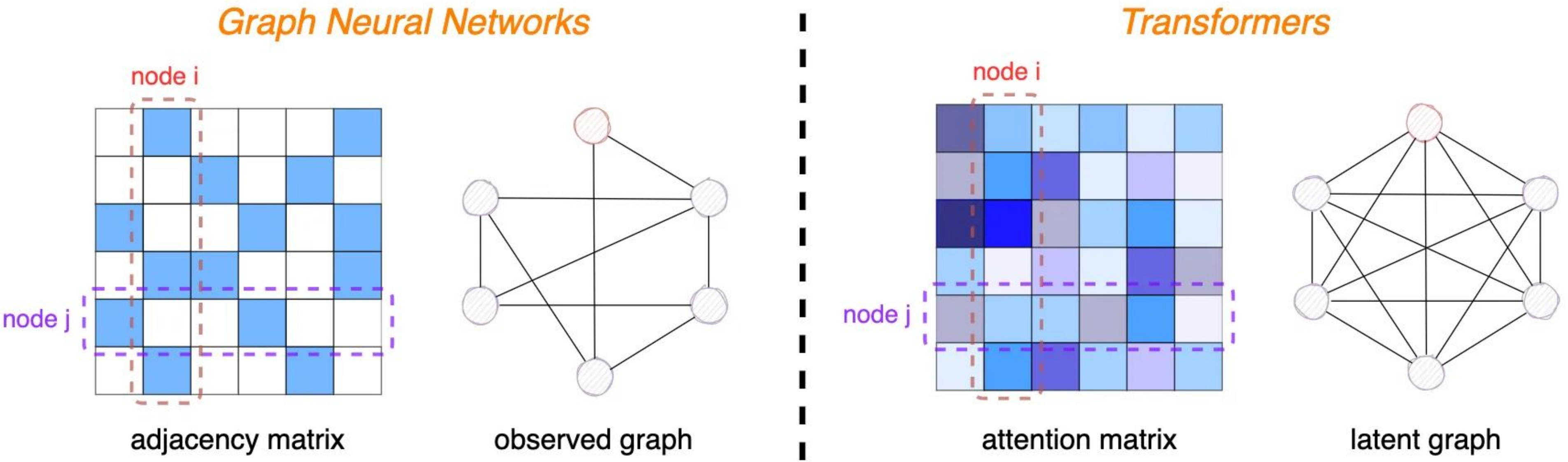


Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. "Machine Learning on Graphs: A Model and Comprehensive Taxonomy." arXiv, April 11, 2022. http://arxiv.org/abs/2005.03675.

Graph Neural Networks

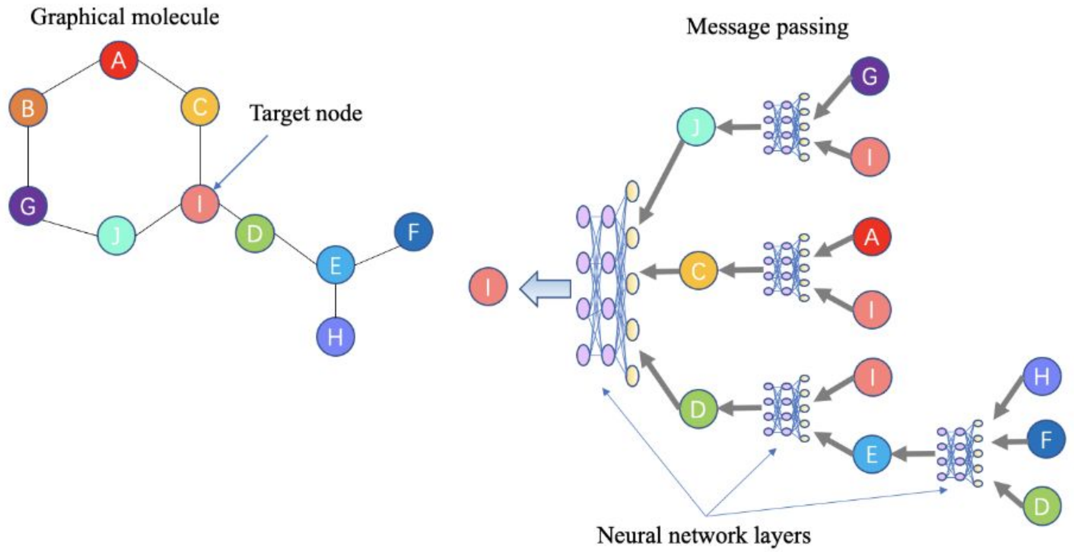
Graph Transformers

Deep embedding: **Graph Transformers**



Wu Q. Medium. 2023 [cited 2024 Nov 22]. How to Build Graph Transformers with O(N) Complexity. Available from: https://towardsdatascience.com/how-to-build-graph-transformers-with-o-n-complexity-d507e103d30a

Deep embedding: **Using node features**



Zhang, Yue, et al. "Application of computational biology and artificial intelligence in drug design." International journal of molecular sciences 23.21 (2022): 13568.

Initialisation

- Random
- Shallow embeddings
- Node features

Aggregation

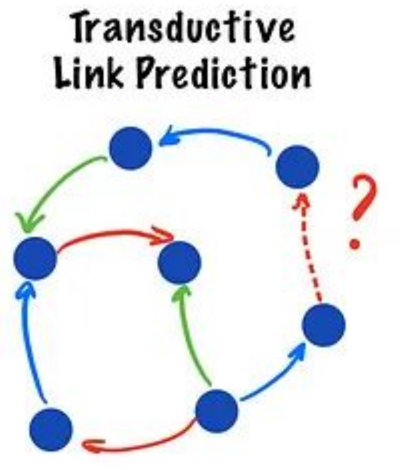
$$m_v^i = A\left(\{s_u^i\}_{u \in N(v)}\right),$$

Update

$$s_v^{i+1} = U\left(s_v^i, m_v^i\right),$$

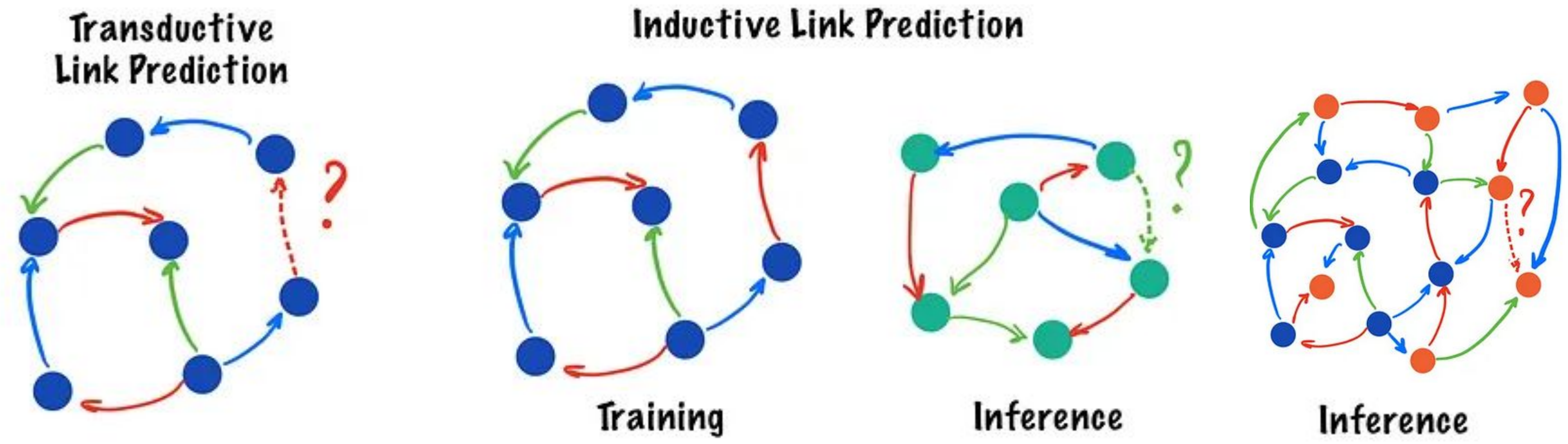
Key concept: **Inductive and Transductive learning**

Galkin M. Medium. 2022 [cited 2024 Nov 22]. Inductive Link Prediction in Knowledge Graphs. Available from: https://towardsdatascience.com/inductive-link-prediction-in-knowledge-graphs-23f249c31961



Key concept: **Inductive and Transductive learning**

Galkin M. Medium. 2022 [cited 2024 Nov 22]. Inductive Link Prediction in Knowledge Graphs. Available from: https://towardsdatascience.com/inductive-link-prediction-in-knowledge-graphs-23f249c31961



- ➢ Transductive models learn node-specific representations for **known nodes**.
- ➢ Inductive models learn **generalizable functions** based on node features and their context.

Training a KG embedding model

Available tools for KG embedding

PyTorch:
Training

TorchKGE:
Shallow embedding

Pytorch Geometric:
Graph Neural Networks

<table>
<tr><th>Framework</th><th>Model Name</th><th>Model Type</th><th>Implementation</th></tr>
<tr><td rowspan="11">Decoders</td><td>TransE</td><td rowspan="5">Translational</td><td rowspan="11">TorchKGE</td></tr>
<tr><td>TransH</td></tr>
<tr><td>TransR</td></tr>
<tr><td>TransD</td></tr>
<tr><td>TorusE</td></tr>
<tr><td>RESCAL</td><td rowspan="5">Bilinear</td></tr>
<tr><td>DistMult</td></tr>
<tr><td>HolE</td></tr>
<tr><td>ComplEx</td></tr>
<tr><td>ANALOGY</td></tr>
<tr><td>ConvKB</td><td>Convolutional</td></tr>
<tr><td rowspan="2">Encoders</td><td>RGCNConv</td><td rowspan="2">Deep</td><td rowspan="2">PyTorch Geometric</td></tr>
<tr><td>RGATConv</td></tr>
</table>

KGE workflow for link prediction

➢ **Splitting the graph triplets** into train, validation, and test sets.

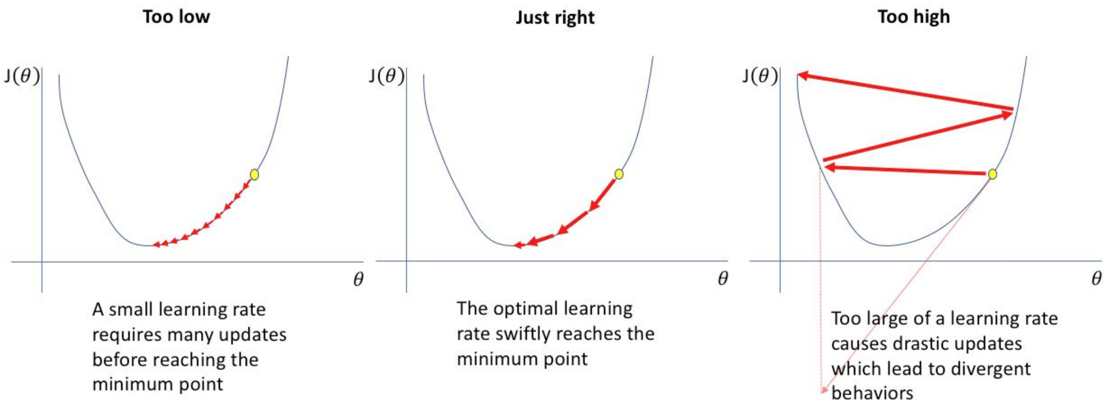
KGE workflow for link prediction

- **Splitting the graph triplets** into train, validation, and test sets.
- **Defining the model**:
    - Choose the model type (TransE, RESCAL, …)
    - Specify the loss function (e.g., margin ranking loss, binary-cross-entropy loss, …).
    - Select the optimizer (e.g., Adam, SGD, ..).

KGE workflow for link prediction

- **Splitting the graph triplets** into train, validation, and test sets.
- **Defining the model**:
  - Choose the model type (TransE, RESCAL, …)
  - Specify the loss function (e.g., margin ranking loss, binary-cross-entropy loss, …).
  - Select the optimizer (e.g., Adam, SGD, ..).
- **Defining the hyperparameters**:
  - Set the embedding dimension.
  - Determine the number of epochs.
  - Specify additional hyperparameters (e.g., learning rate, regularization terms).

Jeremy Jordan [Internet]. 2018 [cited 2024 Nov 22]. Setting the learning rate of your neural network. Available from: https://www.jeremyjordan.me/nn-learning-rate/



KGE workflow for link prediction

- **Splitting the graph triplets** into train, validation, and test sets.
- **Defining the model**:
  - Choose the model type (TransE, RESCAL, …)
  - Specify the loss function (e.g., margin ranking loss, binary-cross-entropy loss, …).
  - Select the optimizer (e.g., Adam, SGD, ..).
- **Defining the hyperparameters**:
  - Set the embedding dimension.
  - Determine the number of epochs.
  - Specify additional hyperparameters (e.g., learning rate, regularization terms).
- **Defining the negative sampling strategy**:
  - Choose how to generate negative samples

KGE workflow for link prediction

- **Splitting the graph triplets** into train, validation, and test sets.
- **Defining the model**:
  - Choose the model type (TransE, RESCAL, …)
  - Specify the loss function (e.g., margin ranking loss, binary-cross-entropy loss, …).
  - Select the optimizer (e.g., Adam, SGD, ..).
- **Defining the hyperparameters**:
  - Set the embedding dimension.
  - Determine the number of epochs.
  - Specify additional hyperparameters (e.g., learning rate, regularization terms).
- **Defining the negative sampling strategy**:
  - Choose how to generate negative samples
- **Defining a quality metric** for evaluating the model on the validation set (e.g., MRR, Hits@K).

KGE workflow for link prediction

Known Facts in the KG:

Paris, city_of, France
Marseille, city_of, France
New York, city_of, USA
Washington, city_of, USA

Élysée, located_in, Paris
White House, located_in, Washington
Broadway, located_in, New York
Eiffel Tower, located_in, Paris
Bonne Mère, located_in, Marseille

Élysée, is_a, Government Building
White House, is_a, Government Building
Broadway, is_a, Cultural Landmark
Eiffel Tower, is_a, Cultural Landmark
Bonne Mère, is_a, Cultural Landmark

Washington, capital_of, USA

Embeddings

Paris
Marseille
New York
Washington
France
USA
Élysée
White House
Broadway
Eiffel Tower
Bonne Mère
Government Building
Cultural Landmark
city_of
located_in
is_a
capital_of

Link Prediction: finding the capital of France

Computing a score for each possible head
(?, capital_of, France)

s(**Paris**, **capital_of**, **France**) = 0.9
s(**Marseille**, **capital_of**, **France**) = 0.7
s(**Washington**, **capital_of**, **France**) = 0.6
s(**New York**, **capital_of**, **France**) = 0.5
s(**Élysée**, **capital_of**, **France**) = 0.4
s(**White House**, **capital_of**, **France**) = 0.2
s(**USA**, **capital_of**, **France**) = 0.2
etc…

with s(**h, r, t**), a model-dependant scoring function

KGE workflow for link prediction

Known Facts in the KG:

Paris, city_of, France
Marseille, city_of, France
New York, city_of, USA
Washington, city_

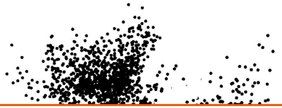
Élysée, located_ir
White House, loc
Broadway, locate
Eiffel Tower, loca
Bonne Mère, loca

Élysée, is_a, Gove
White House, is_a
Broadway, is_a, C
Eiffel Tower, is_a,
Bonne Mère, is_a, Cultural Landmark

Washington, capital_of, USA

Embeddings

Paris
Marseille



is_a
capital_of

Link Prediction: finding the capital of France

Computing a score for each possible head
(?, capital_of, France)

= 0.9
nce) = 0.7
rance) = 0.6
nce) = 0.5
) = 0.4
France) = 0.2
= 0.2

ndant scoring function

Link Prediction tasks evaluated with metric

Mean Reciprocal Rank

$$\mathrm{MRR} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\mathrm{rank}_i}$$

KGE workflow for link prediction

- **Splitting the graph triplets** into train, validation, and test sets.
- **Defining the model**:
    - Choose the model type (TransE, RESCAL, …)
    - Specify the loss function (e.g., margin ranking loss, binary-cross-entropy loss, …).
    - Select the optimizer (e.g., Adam, SGD, ..).
- **Defining the hyperparameters**:
    - Set the embedding dimension.
    - Determine the number of epochs.
    - Specify additional hyperparameters (e.g., learning rate, regularization terms).
- **Defining the negative sampling strategy**:
    - Choose how to generate negative samples
- **Defining a quality metric** for evaluating the model on the validation set (e.g., MRR, Hits@K).

$$\mathrm{MRR} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{\mathrm{rank}_i}$$

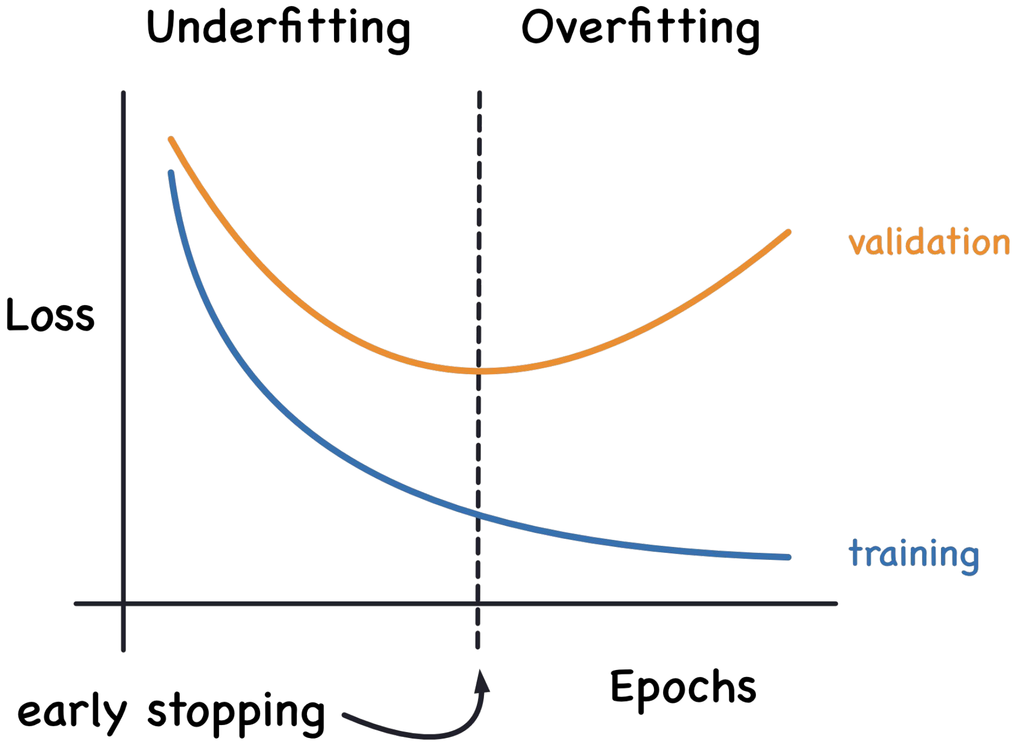
KGE workflow for link prediction

- **Splitting the graph triplets** into train, validation, and test sets.
- **Defining the model**:
    - Choose the model type (TransE, RESCAL, …)
    - Specify the loss function (e.g., margin ranking loss, binary-cross-entropy loss, …).
    - Select the optimizer (e.g., Adam, SGD, ..).
- **Defining the hyperparameters**:
    - Set the embedding dimension.
    - Determine the number of epochs.
    - Specify additional hyperparameters (e.g., learning rate, regularization terms).
- **Defining the negative sampling strategy**:
    - Choose how to generate negative samples
- **Defining a quality metric** for evaluating the model on the validation set (e.g., MRR, Hits@K).
- **Training the model**, and monitoring:
    - Monitoring performances on the training and validation sets

Overfitting and Underfitting [Internet]. [cited 2024 Nov 22]. Available from: https://kaggle.com/code/ryanholbrook/overfitting-and-underfitting

KGE workflow for link prediction

- **Splitting the graph triplets** into train, valic
- **Defining the model**:
  - Choose the model type (TransE, RI
  - Specify the loss function (e.g., marg
  - Select the optimizer (e.g., Adam, S(
- **Defining the hyperparameters**:
  - Set the embedding dimension.
  - Determine the number of epochs.
  - Specify additional hyperparameters
- **Defining the negative sampling strategy**
  - Choose how to generate negative s
- **Defining a quality metric** for evaluating th
- **Training the model**, and monitoring:
  - Monitoring performances on the tra



Overfitting and Underfitting [Internet]. [cited 2024 Nov 22]. Available from: https://kaggle.com/code/ryanholbrook/overfitting-and-underfitting

KGE workflow for link prediction

- **Splitting the graph triplets** into train, validation, and test sets.
- **Defining the model**:
  - Choose the model type (TransE, RESCAL, …)
  - Specify the loss function (e.g., margin ranking loss, binary-cross-entropy loss, …).
  - Select the optimizer (e.g., Adam, SGD, ..).
- **Defining the hyperparameters**:
  - Set the embedding dimension.
  - Determine the number of epochs.
  - Specify additional hyperparameters (e.g., learning rate, regularization terms).
- **Defining the negative sampling strategy**:
  - Choose how to generate negative samples
- **Defining a quality metric** for evaluating the model on the validation set (e.g., MRR, Hits@K).
- **Training the model**, and monitoring:
  - Monitoring performances on the training and validation sets
- **Evaluating the model**:
  - Assess the model's performance on the test set using the defined quality metrics**.**
    - **Good enough? Inference!**
    - **Not good enough? Back to step 2!**

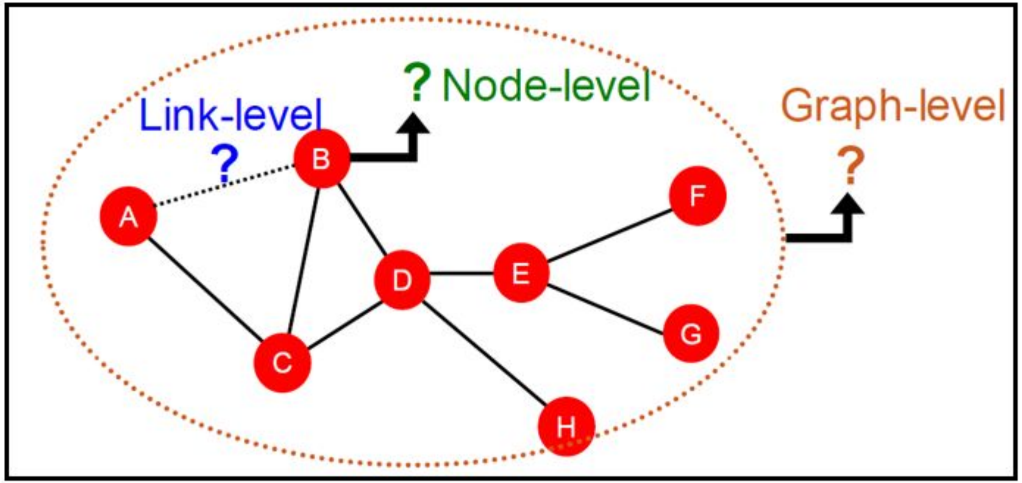
KGE workflow for link prediction

- **Splitting the graph triplets** into train, validation, and test sets.
- **Defining the model**:
  - Choose the model type (TransE, RESCAL, …)
  - Specify the loss function (e.g., margin ranking loss, bin
  - Select the optimizer (e.g., Adam, SGD, ..).
- **Defining the hyperparameters**:
  - Set the embedding dimensi
  - Determine the num
  - Specify ad
- **Definin**
  -
- **Defini** alidation set (e.g., MRR, Hits@K).
- **Trainin**
  - I aining and validation sets
- **Evaluati**
  - As ers performance on the test set using the defined quality metrics**.**
    - **ood enough? Inference!**
    - **Not good enough? Back to step 2!**

Hands-on on Wednesday!!
Don't forget to install your conda environments!

Applications in Biomedicine

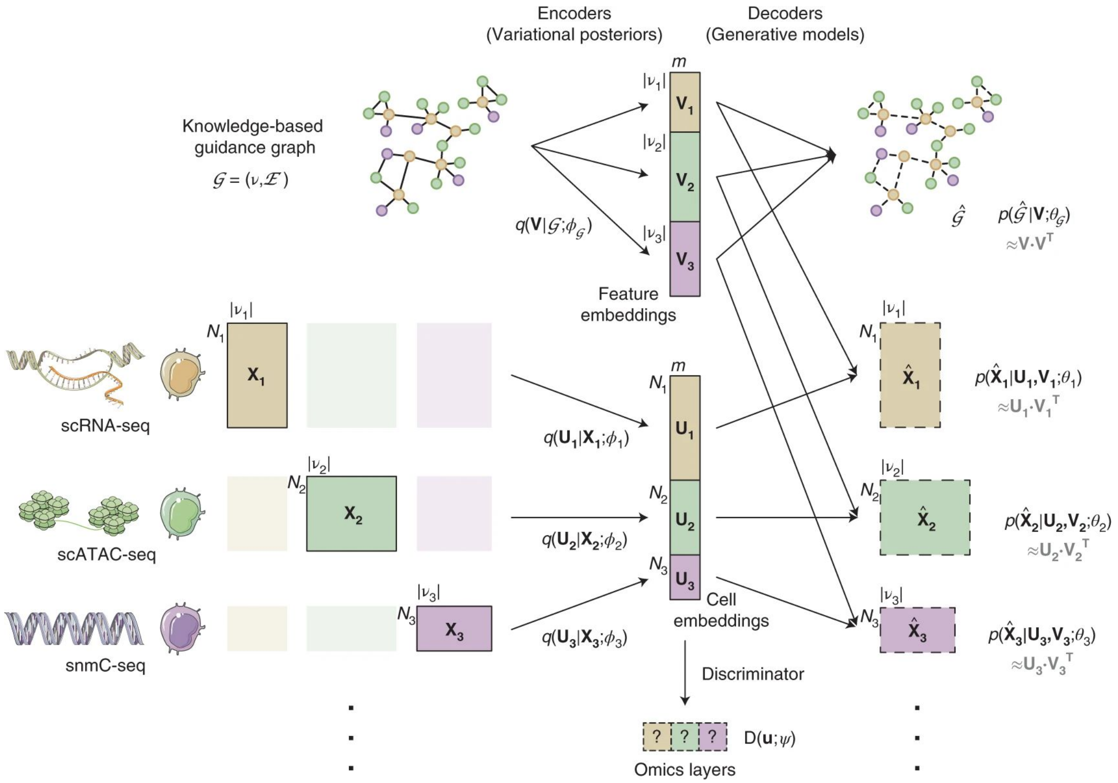
Tasks in Knowledge Graph Embedding



- **Classification**
  - Posititive/negative triplet
  - Node properties
  - Graph properties
- **Clustering**
  - Node clustering
  - Graph clustering
  - (Edge clustering)
- **Regression**
  - Node properties
  - Edge weights
  - Graph properties

Ajay Taneja, Extracting Node Level Features from Graph Networks for Machine Learning Models, LinkedIn post

Towards Graph Learning for omics data integration?



Cao, ZJ., Gao, G. Multi-omics single-cell data integration and regulatory inference with graph-linked embedding. Nat Biotechnol 40, 1458–1466 (2022). https://doi.org/10.1038/s41587-022-01284-4