Graph Neural Networks for Graph Query Matching & Graphs Isomorphism Counting

lecture-10

Course on Graph Neural Networks (Winter Term 21/22)

Prof. Dr. Holger Giese (holger.giese@hpi.de)

Christian Medeiros Adriano (christian.adriano@hpi.de) - **"Chris"**

Matthias Barkowski (matthias.barkowski@hpi.de)

Hasso Plattner Institut

IT Systems Engineering | Universität Potsdam

Intro

Graph Isomorphisms consists of **counting** the number of subgraphs in graph database that present the same configuration of nodes and edges.

Graph Isomorphism
Counting

Graph Matching consists of finding the number of graphs that match a given **query**, which is represented by a graph.

Graph Query **Matching**

Two fundamental challenges of GNNs

Expressivity = repeated neighbor mixing collapses embeddings of different nodes into a fixed low-dimensional subspace.

Scalability = recursive expansion of neighbor nodes results in exponentially growing neighborhood size.

Graph Query Matching

Goal: find out the query graph from the data graph, i.e., it is a search problem that involves ranking results (matches)

Neuro Subgraph Matching [Ying 2020]

Limitations of previous work (Bai et al., 2019; Li et al., 2019; Xu et al., 2019):

embed graphs into vector spaces

→ do not impose geometric structure in the embedding space.

→

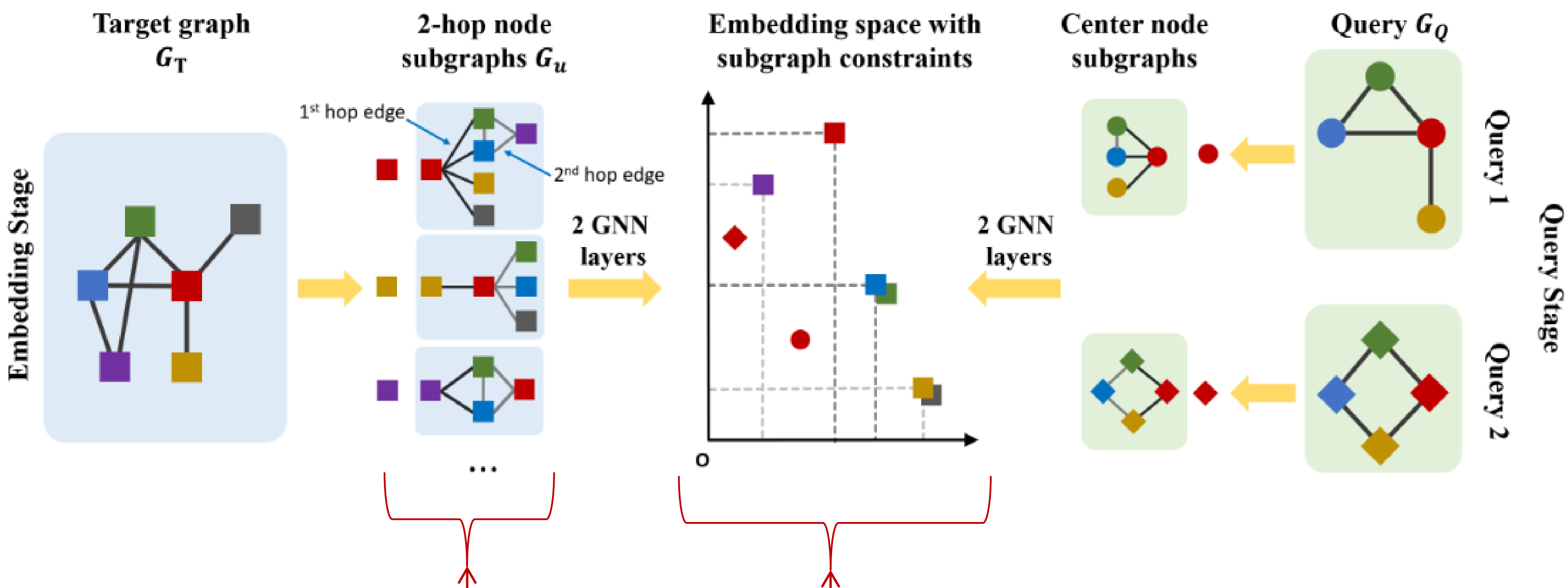
Conversely, using a GNN creates an ordered embedding that

→leads to a well-structured embedding space

→to efficiently navigate it in order to find subgraphs as well as supergraph

source: Ying, R., Lou, Z., You, J., Wen, C., Canedo, A., & Leskovec, J. (2020). Neural Subgraph Matching

Framework of Neuro Subgraph Matching [Ying 2020]



GNN is trained to learn the order embedding

Max-margin loss is used to ensure that the subgraph relationships are captured

source: Ying, R., Lou, Z., You, J., Wen, C., Canedo, A., & Leskovec, J. (2020). Neural Subgraph Matching

Decision problems of subgraph matching

Problem-1 Matching query to datasets

Given a target graph G_T and a query G_Q, predict if G_Q is isomorphic to a subgraph of G_T

Then uses a neural model to decompose Problem 1 and solve Problem-2 (with a certain accuracy)

Problem-2 Matching neighborhoods

Given a neighborhood G_u around node u and query G_Q anchored at node q, make binary prediction of whether G_q is a subgraph of G_u where node q corresponds to u

Caveat - GNNs are good at similarity measurement tasks but are not good at substructure extraction tasks: shortest path, subgraph, self-loop, etc.

References Graph Query Matching

1. Ying, Z., Wang, A., You, J., Wen, C., Canedo, A., & Leskovec, J. (2020). **Neural subgraph matching**.
2. Lan, Z., Yu, L., Yuan, L., Wu, Z., Niu, Q., & Ma, F. (2021). **Sub-GMN: The Subgraph Matching Network Model**. arXiv preprint arXiv:2104.00186.
3. Sarlin, P. E., DeTone, D., Malisiewicz, T., & Rabinovich, A. (2020). **Superglue: Learning feature matching with graph neural networks.** In Proceedings of the IEEE/CVF, pp. 4938-4947. Data and code are available at https://github.com/magicleap/SuperGluePretrainedNetwork
4. Krleža, D., & Fertalj, K. (2016). **Graph matching using hierarchical fuzzy graph neural networks**. IEEE Transactions on Fuzzy Systems, 25(4), 892-904.
5. Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., ... & Sun, M. (2020). **Graph neural networks: A review of methods and applications**. AI Open, 1, 57-81.
6. your suggestion?

Graph Isomorphism Counting

Goal: find the number of query graphs (isomorphism, graph structures) in the data graph.

Different Settings for Counting

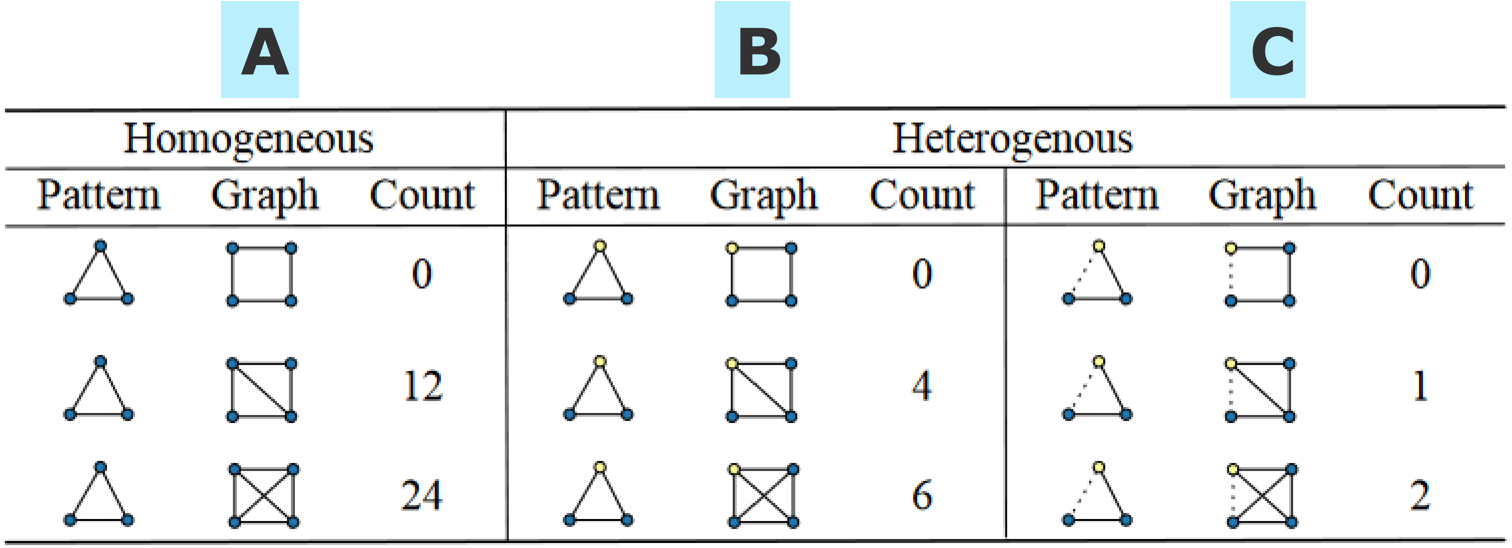
A. Homogeneous graphs

B. Heterogeneous vertex graphs

- two types of nodes

C. Heterogeneous vertex and heterogeneous edge graphs

- two types of nodes and two types of edges



source: Liu et al., 2020, Neural Subgraph Isomorphism Counting, KDD '20

Intuition of the Graph Counting Problem

The counting problem can be formulated as a question-answering problem well established in natural language processing, but with slight difference in the outcome:

- Question Answering Model (Text Queries):
  - answers are facts extracted from the provided texts

- Graph Counting Model:
  - answers are summary statistics of matched local patterns.

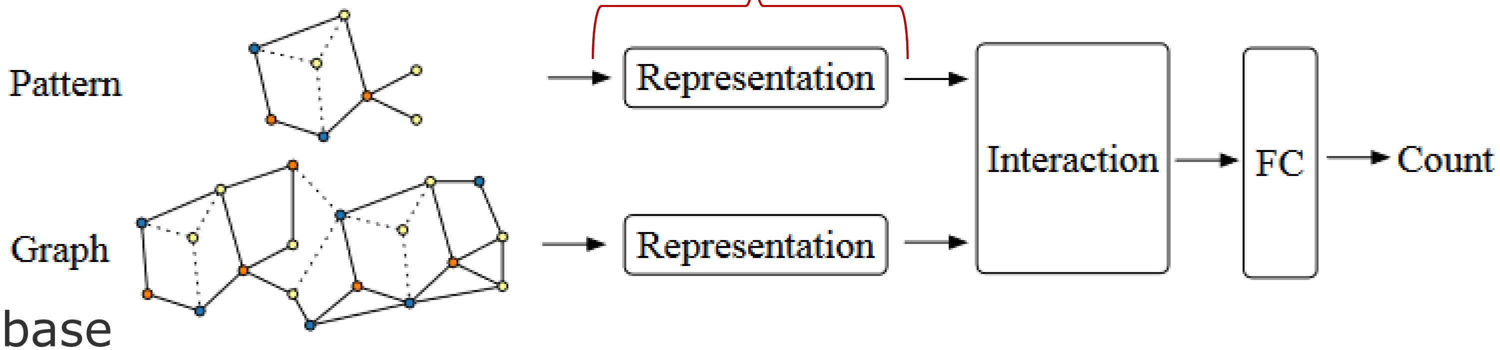
source: Liu et al., 2020, Neural Subgraph Isomorphism Counting, KDD '20

Framework of neural subgraph isomorphism counting models [Liu 2020]

Encoding makes the problem learnable.

Caveat: Learning based approaches are inexact

Query



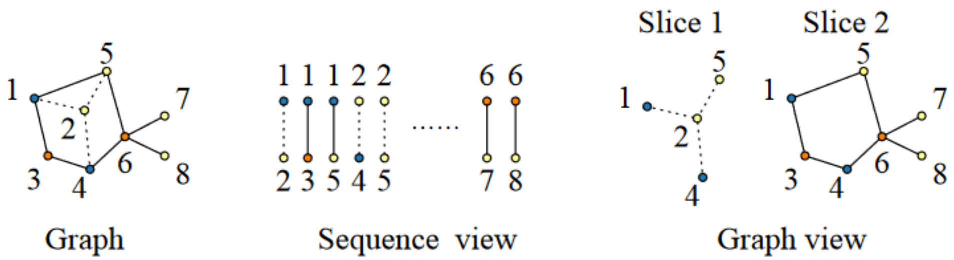
Graph database

Encoding the graph (or the pattern) should be either represented:

A. sequence of edges, **or**
B. a series of adjacent matrices and vertex features.

A

B



source: Liu et al., 2020, Neural Subgraph Isomorphism Counting, KDD '20

Results [Liu et al. 2020]

Generalized to count large patterns and data graphs in linear time compared to the exponential time of the original NP-complete problem.

Experimental results showed that learning based subgraph isomorphism counting can speed up the traditional algorithm (VF2) in up to 1000 times with acceptable errors.

Data and code are available at https://github.com/HKUST-KnowComp/NeuralSubgraphCounting

References Graph Isomorphism Counting

1. Liu, X., Pan, H., He, M., Song, Y., Jiang, X., & Shang, L. (2020, August**). Neural subgraph isomorphism counting**. In Proceedings of the 26th ACM SIGKDD, pp. 1959-1969, Data and code are available at https://github.com/HKUST-KnowComp/NeuralSubgraphCounting
2. Bouritsas, G., Frasca, F., Zafeiriou, S., & Bronstein, M. M. (2020**). Improving graph neural network expressivity via subgraph isomorphism counting**. arXiv preprint arXiv:2006.09252
3. Chen, Z., Chen, L., Villar, S., & Bruna, J. (2020). Can graph neural networks count substructures?. arXiv preprint arXiv:2002.04025.
4. Tahmasebi, B., Lim, D., & Jegelka, S. (2020). **Counting substructures with higher-order graph neural networks: Possibility and impossibility results**. arXiv preprint arXiv:2012.03174.
5. You, J., Gomes-Selman, J., Ying, R., & Leskovec, J. (2021). **Identity-aware graph neural networks**. arXiv preprint arXiv:2101.10320.
6. … your suggestion?

Next tasks

1. Choose one or two papers to read.
2. Write a gist with 4 lines for each item below:
   - Problem (what is it, why is it important, what were the challenges)
   - Approach (encoding, loss function, message-passing-aggregation/algorithm)
   - Evaluation (baseline graph models, metrics, competing algorithms)
3. Share your comments on Slack before the lecture
   - #sota-graph-query-matching
   - #sota-graph-isomorphism-counting

4. Present next Wednesday
   - decide the group topic and
   - start narrowing the specific approach

Advanced Topics

End

GNNs and Weisfeiler-Lehman Isomorphism Test

4- Weisfeiler and lehman go topological: Message passing simplicial networks, 2021 https://www.hpi.uni-potsdam.de/giese/bibadmin/show.php?id=17130

5- The Power of the Weisfeiler-Leman Algorithm for Machine Learning with Graphs, 2021

https://www.hpi.uni-potsdam.de/giese/bibadmin/show.php?id=17134

6- word2vec, node2vec, graph2vec, x2vec: Towards a theory of vector embeddings of structured data, 2020

https://www.hpi.uni-potsdam.de/giese/bibadmin/show.php?id=17149

Querying with GNNs

7- The expressive power of graph neural networks as a query language, 2020

https://www.hpi.uni-potsdam.de/giese/bibadmin/show.php?id=17147

8- Representing Schema Structure with Graph Neural Networks for Text-to-SQL Parsing, 2019

https://www.hpi.uni-potsdam.de/giese/bibadmin/show.php?id=17146

9- The relational data borg is learning

https://www.hpi.uni-potsdam.de/giese/bibadmin/show.php?id=17148

Equivariance and Expressiviness of GNNs

10- Expressive power of invariant and equivariant graph neural networks, 2020

https://www.hpi.uni-potsdam.de/giese/bibadmin/show.php?id=17139

11- Equivariant Subgraph Aggregation Networks, 2021

https://www.hpi.uni-potsdam.de/giese/bibadmin/show.php?id=17136

Search Plan Generation

https://link.springer.com/chapter/10.1007/11841883_27 (static search plans)

https://link.springer.com/article/10.1007/s10270-013-0372-2 (static search plans)

https://journal.ub.tu-berlin.de/eceasst/article/view/268 (dynamic search plans)

https://link.springer.com/chapter/10.1007/978-3-030-23611-3_13 (hybrid search plans; heuristic used in our tool)

LDBC Social Network Benchmark

https://dl.acm.org/doi/pdf/10.1145/2723372.2742786 (initial paper)

https://arxiv.org/abs/2001.02299 (full documentation)

https://github.com/ldbc/ldbc_snb_interactive (repo)

Best regards,

Matthias