Neural Graph Databases

ETH *zürich*

Maciej Besta, Patrick Iff, Florian Scheidl, Kazuki Osawa,
Nikoli Dryden, Michal Podstawski, Tiancheng Chen, Torsten Hoefler
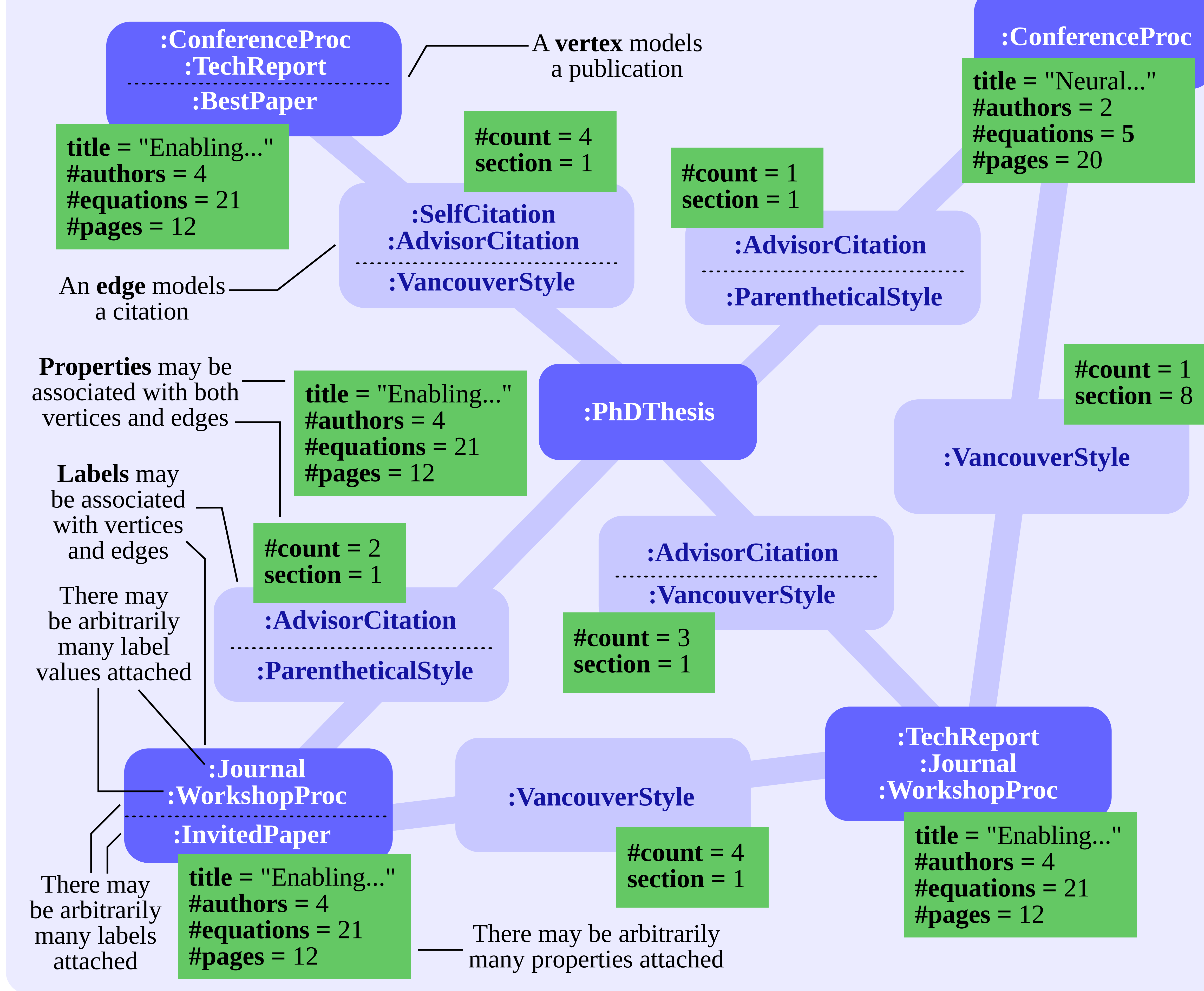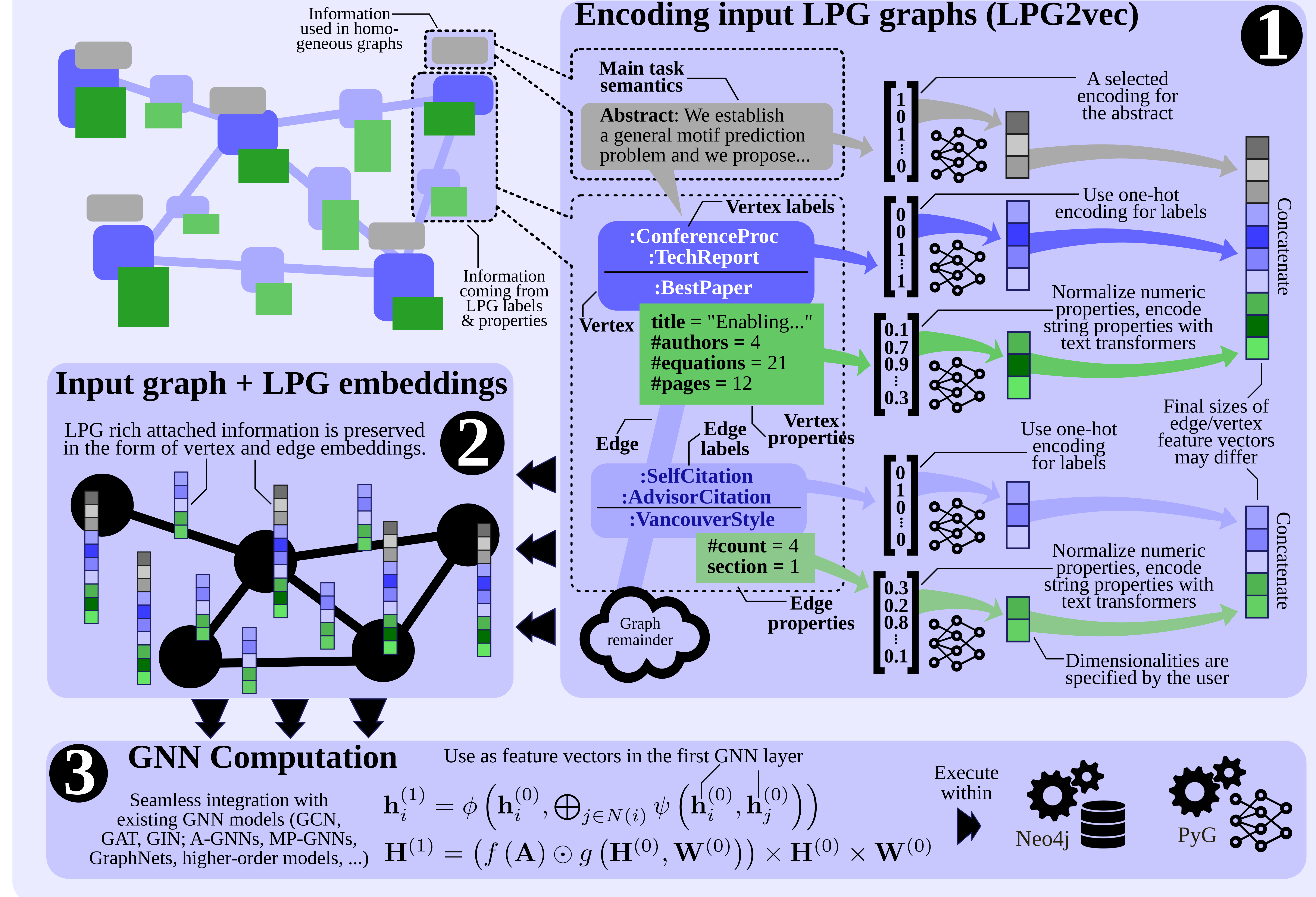Department of Computer Science, ETH Zurich

Motivation

Graph databases (GDBs) are a class of systems heavily used in the industry. A plethora of such systems exist, e.g.,
Neo4j, JanusGraph, Azure Cosmos DB, Amazon Neptune, Virtuoso, ArangoDB, OrientDB, GraphDB, TigerGraph,
GraphDB, Fauna, Dgraph, AllegroGraph, NebulaGraph, Blazegraph, Graph Engine, and many more.
An established data model used in the majority of graph databases is called the Labeled Property Graph (LPG).
How to seamlessly enable the learning capabilities of Graph Neural Networks (GNNs) with GDBs?

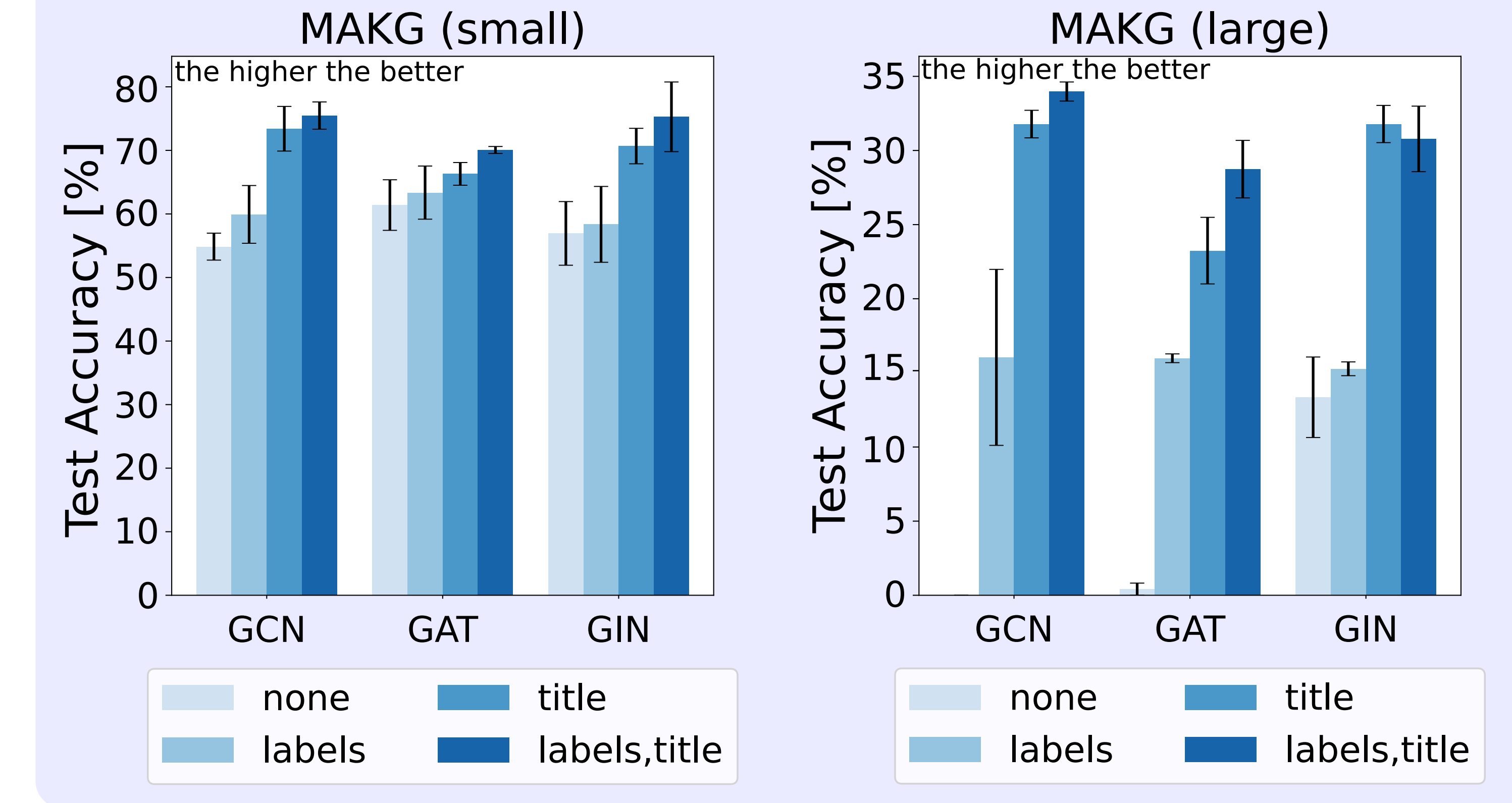## Example Labeled Property Graph (LPG)

Labels in this example:
➼ **Publication type** (a journal paper, a conference paper, a workshop paper, a report, a thesis)
➼ **Publication highlight** (best paper, distinguished paper, invited paper)
➼ **Citation variant** (self citation, advisor citation)
➼ **Citation style** (Vancouver, Parenthetical)

A **vertex** models a publication

An **edge** models a citation

Properties may be associated with both vertices and edges

Labels may be associated with vertices and edges

There may be arbitrarily many label values attached

There may be arbitrarily many labels attached

There may be arbitrarily many properties attached

The Design of LPG2vec: Marrying Graph Databases and Graph Neural Networks

Encoding input LPG graphs (LPG2vec) ❶

Information used in homogeneous graphs

Main task semantics

Abstract: We establish a general motif prediction problem and we propose...

A selected encoding for the abstract

Vertex labels

:ConferenceProc
:TechReport
:BestPaper

Use one-hot encoding for labels

title = "Enabling..."
#authors = 4
#equations = 21
#pages = 12

Normalize numeric properties, encode string properties with text transformers

Information coming from LPG labels & properties

Vertex

Vertex properties

Edge
Edge labels

:SelfCitation
:AdvisorCitation
:VancouverStyle

Use one-hot encoding for labels

#count = 4
section = 1

Final sizes of edge/vertex feature vectors may differ

Normalize numeric properties, encode string properties with text transformers

Concatenate

Graph remainder

Edge properties

Dimensionalities are specified by the user

Input graph + LPG embeddings ❷

LPG rich attached information is preserved in the form of vertex and edge embeddings.

❸ GNN Computation

Use as feature vectors in the first GNN layer

Seamless integration with existing GNN models (GCN, GAT, GIN; A-GNNs, MP-GNNs, GraphNets, higher-order models, ...)

$$\mathbf{h}_i^{(1)} = \phi\left(\mathbf{h}_i^{(0)}, \bigoplus_{j \in N(i)} \psi\left(\mathbf{h}_i^{(0)}, \mathbf{h}_j^{(0)}\right)\right)$$

$$\mathbf{H}^{(1)} = \left(f(\mathbf{A}) \odot g\left(\mathbf{H}^{(0)}, \mathbf{W}^{(0)}\right)\right) \times \mathbf{H}^{(0)} \times \mathbf{W}^{(0)}$$

Execute within

Neo4j   PyG

Node classification

MAKG (small)
the higher the better

Test Accuracy [%]

GCN   GAT   GIN

MAKG (large)
the higher the better

Test Accuracy [%]

GCN   GAT   GIN

none   title
labels   labels,title

Node regression

Neo4j (citations)
the lower the better

Test MAE

GCN   GAT   GIN

none   year
labels   labels,year

Neo4j (Twitter trolls)
the lower the better

Test MAE

GCN   GAT   GIN

none   follower_cnt
labels   labels,follower_cnt