Molecular Graphs and GNNs

Oct 24, 2024

Common molecular representations

- Molecular representations allow encoding chemical structures in a format that can be processed by computers for applications in drug discovery, cheminformatics, and computational chemistry.
- **Key Representations:**
  - SMILES (Simplified Molecular Input Line Entry System)
  - SMARTS (SMILES Arbitrary Target Specification)
  - SELFIES (Self-Referencing Embedded Strings)
  - InChI (International Chemical Identifier)
  - Molecular graphs

SMILES (Simplified Molecular Input Line Entry System)

- **Definition:** A linear text-based representation of chemical structures using ASCII characters.
- **Key Features:**
  - Encodes atoms, bonds, and connectivity in a compact string.
  - Parentheses denote branching, and numbers indicate ring closures.
- **Example:** Ethanol → CCO
- **Advantages:** Widely used, compact, human-readable.
- **Limitations:** Complex structures (e.g., stereochemistry, tautomers) can be hard to encode.

SMARTS (SMILES Arbitrary Target Specification)

- **Definition:** A pattern matching language based on SMILES, used to define substructures and perform searches in molecular databases.
- **Key Features:**
  - Allows encoding more complex molecular features like atom classes, aromaticity, and ring systems.
  - Commonly used in cheminformatics tools for substructure searching.
- **Example:** Aromatic ring → c1ccccc1
- **Advantages:** Flexible for substructure searches.
- **Limitations:** More complex and less intuitive than SMILES.

SELFIES (Self-Referencing Embedded Strings)

- **Definition:** A molecular representation that is a fully reversible string-based encoding designed to always produce valid chemical structures.
- **Key Features:**
  - SELFIES strings are robust and can always be decoded into valid molecular structures, even when mutated or modified.
- **Example:** Benzene → C1=CC=CC=C1
- **Advantages:** Highly error-tolerant; all SELFIES strings decode to valid molecules, making it useful in machine learning.
- **Limitations:** Less human-readable than SMILES.

InChI (International Chemical Identifier)

- **Definition:** A standardized textual identifier developed by IUPAC that provides a unique, canonical representation of a molecule.
- **Key Features:**
  - Includes layers for different molecular features: connectivity, charge, stereochemistry, and isotopic composition.
- **Example:** Ethanol → InChI=1S/C2H6O/c1-2-3/h3H,2H2,1H3
- **Advantages:** Standardized, unique identifier for each molecule.
- **Limitations:** Less compact and readable than SMILES, more difficult to work with in some applications.

Molecular Graphs

- **Definition:** A graph-based representation where atoms are vertices, and bonds are edges.
- **Key Features:**
  - Useful in cheminformatics and computational chemistry for visualizing molecular structures.
  - Can be represented in adjacency matrices or graph notations.
- **Example:** Atoms are nodes, and bonds are lines connecting them.
- **Advantages:** Intuitive for understanding connectivity.
- **Limitations:** Not as compact as string-based representations.

Comparison of Molecular Representations

Representation	Description	Advantages	Limitations
SMILES	Linear ASCII string	Compact, widely used	Complex to encode stereochemistry
SMARTS	Substructure search patterns	Powerful for substructure matching	Less intuitive, more complex
SELFIES	Robust string encoding	Always valid molecules, error-tolerant	Less human-readable
InChI	Canonical identifier	Standardized, unique	Less compact, harder to use directly
Molecular Graphs	Graph-based, visual representation	Intuitive for structure visualization	Not as compact, harder to manipulate

SMILES notation - 5 rules

1. **Atoms & bonds**
   Represented by their chemical symbol (Na,C,F)
   Lower case - aromatic (c1ccccc1)

   - Single bond
   = Double bond
   # Triple bond
   * Aromatic bond
   . Disconnected structures

SMILES notation - 5 rules

2. Simple chains

Hydrogens are usually suppressed

CC		CH3CH3	Ethane
C=C	CH2CH2		Ethene
CBr		CH3Br	Bromomethane
C#N	C=N	Hydrocyanic acid	
Na.Cl	NaCl	Sodium chloride	

SMILES notation - 5 rules

3. Branches

Branches are specified by parentheses, connect to preceding atom
Includes bond character after left parenthesis if needed

CC(O)C	2-Propanol
CC(=O)C	2-Propanone
CC(CC)C	2-Methylbutane
CC(C)CC(=O)	2-Methylbutanal
c1c(N(=O)=O)cccc1	Nitrobenzene
CC(C)(C)CC	2,2-Dimethylbutane

SMILES notation - 5 rules

4. Rings

Rings are specified by numbers. Same number indicates opening/closing atoms of ring. Bond type comes after atom but before number.

C=1CCCCC1	Cyclohexene
C*1*C*C*C*C*C1	Benzene
c1ccccc1	Benzene
C1OC1CC	Ethyloxirane
c1cc2ccccc2cc1	Naphthalene

SMILES notation - 5 rules

5. Charges
Charge is specified in {}

CCC(=O)O{-1} Ionized form of propanoic acid

CCC(=O)O{-}

c1ccccn{+1}1CC(=O)O 1-Carboxylmethyl pyridinium

c1cc2ccccc2cc1 Naphthalene

SMILES notation

Ambiguous names
[] used to separate individual atoms

Sc → sulfur + aromatic carbon

[Sc] → Scandium

Morgan Fingerprints

Definition: Morgan fingerprints (also known as circular fingerprints) are a type of molecular descriptor used to encode chemical structures for computational analysis in cheminformatics and drug discovery.

Key Use: Morgan fingerprints are widely used for tasks such as similarity searching, quantitative structure-activity relationship (QSAR) modeling, and virtual screening.

Origin: They are based on the Morgan algorithm, an extension of the Extended-Connectivity Fingerprints (ECFP).

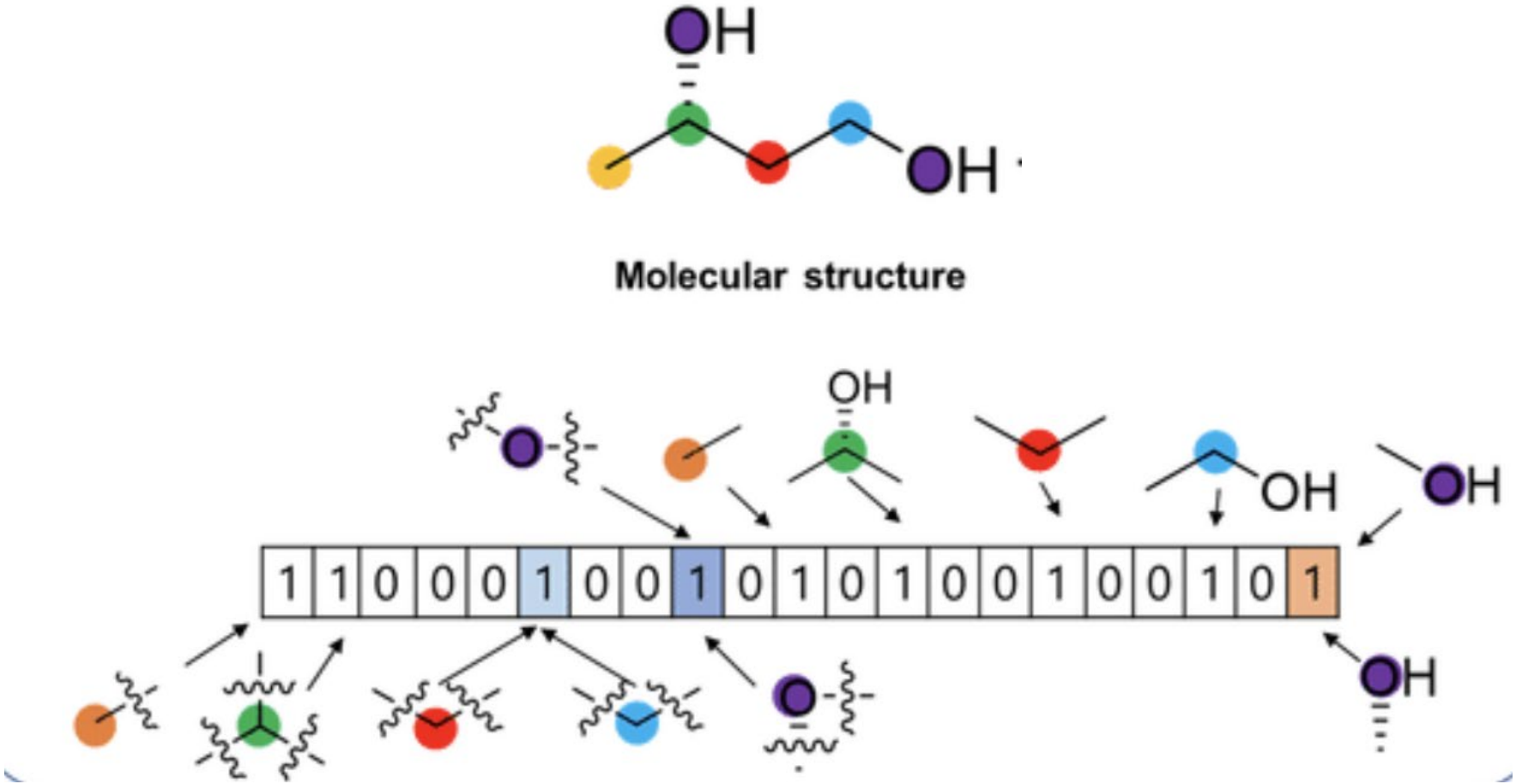
How Morgan Fingerprints Work

- **Circular Representation:**
  - Atoms in the molecule are used as centers, and a circular substructure around each atom is encoded.
  - The radius defines the size of the substructure around the central atom that will be considered.
- **Feature Encoding:**
  - Each circular substructure is encoded as a binary vector (bit string), where the presence of specific features (e.g., atoms, bonds, rings) is recorded as 1, and absence is recorded as 0.
  - The radius determines how many layers of atoms around the central atom are considered (e.g., radius 2 includes atoms and bonds up to two bonds away).
- **Collision of Substructures:**
  - Substructures are hashed into a fixed-length fingerprint (e.g., 1024 bits) where hash collisions may occur, meaning multiple substructures can map to the same bit.

Example of Morgan Fingerprint Generation

- **Step 1:** Start with an atom in the molecule.
- **Step 2:** Build a substructure by expanding outward to a specified radius (e.g., radius 2).
- **Step 3:** Hash the substructure into a unique identifier (bit).
- **Step 4:** Repeat for all atoms, and generate a binary fingerprint representing the entire molecule.
- **Example benzene (C6H6):**
  - Atoms in a ring are encoded by circular substructures at a certain radius.
  - These substructures are hashed into binary bits.

Example



Kim et al., 2020 10.26434/chemrxiv.12885494

Advantages of Morgan Fingerprints

- **High Sensitivity to Structural Changes:**
  - Captures local chemical environments, making them sensitive to small structural modifications, which is useful for fine-tuning molecular features.
- **Efficiency:**
  - Computationally efficient and easy to implement, even for large compound libraries.
- **Widely Used:**
  - Supported in many cheminformatics tools (e.g., RDKit) and extensively used in QSAR, virtual screening, and similarity searching.

Applications of Morgan Fingerprints

- **Molecular Similarity Searches:**
  - Used to identify molecules with similar substructures to a query molecule, aiding in lead optimization and compound screening.
- **QSAR Models:**
  - Morgan fingerprints are often used as descriptors in machine learning models to predict biological activity based on chemical structure.
- **Virtual Screening:**
  - Widely used in virtual compound screening to find potential drug candidates that share structural similarity with known active molecules.

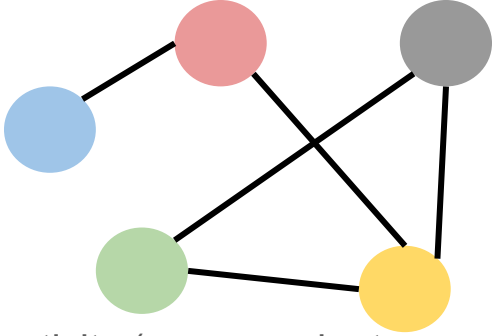
Overview of Molecular Graphs

- **Definition:** A molecular graph is a mathematical representation of a molecule where atoms are represented as nodes (vertices), and bonds between atoms are represented as edges (lines).
- **Key Concept:** Molecular graphs are used to model the structure and connectivity of molecules in computational chemistry and cheminformatics.
- **Applications:** Used in chemical similarity searching, QSAR modeling, and structure-based drug design.

Elements of a Molecular Graph

- **Nodes (Vertices):**
  - Represent individual atoms in the molecule.
  - Each node is labeled with the type of atom (e.g., carbon, oxygen).
- **Edges (Lines):**
  - Represent bonds between atoms.
  - Can represent single, double, or triple bonds, and may be labeled accordingly.
- **Weighted Edges (optional):**
  - In some molecular graphs, edges may be weighted to represent bond strength or bond order.
- **Example:**
  - Ethanol (CH_3CH_2OH):
  - **Nodes:** C, H, O atoms.
  - **Edges:** Single bonds connecting C-C, C-H, C-O, and O-H.

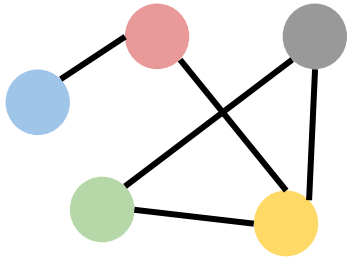
Types of Graph



- **Undirected Graph:**
  - Edges have no direction, representing simple atom-to-atom connectivity (e.g., covalent bonds).
  - Most common representation for small molecules.
- **Directed Graph:**
  - Edges have direction, used to represent specific flow or orientation (e.g., in reaction mechanisms).
- **Labeled Graph:**
  - Both nodes and edges are labeled to reflect atom types and bond orders.
  - Useful in applications where chemical details (atom type, bond type) are important.
- **Weighted Graph:**
  - Edges or nodes are assigned weights (numerical values) to indicate properties like bond length or bond strength.

Representation of Molecular Graphs

- **Adjacency Matrix:**
  - A matrix where rows and columns represent atoms, and the presence of a bond between two atoms is indicated by 1 in the corresponding cell (or by the bond order).
- **Adjacency List:**
  - A list where each atom is paired with the atoms to which it is connected, used to describe the graph's connectivity more efficiently.
- **Graph Visualization:**
  - Graphical depictions where atoms (nodes) are connected by lines (edges), showing the molecule's structure.



Applications of molecular graphs in drug discovery

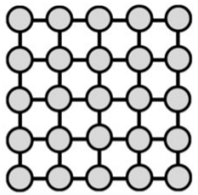
- **Chemical Similarity Searching:**
  - Molecular graphs are used to compare structural similarity between molecules, aiding in lead identification and compound screening.
- **QSAR Models:**
  - Graph-based features (e.g., connectivity indices, topological descriptors) are used as inputs for Quantitative Structure-Activity Relationship (QSAR) models to predict biological activity.
- **Graph Neural Networks (GNNs):**
  - Used in machine learning, where molecules are treated as graphs and deep learning models predict properties like toxicity or binding affinity.
- **Structure-Based Drug Design:**
  - Molecular graphs can help visualize and analyze binding sites in proteins, assisting in designing drugs with specific interactions.

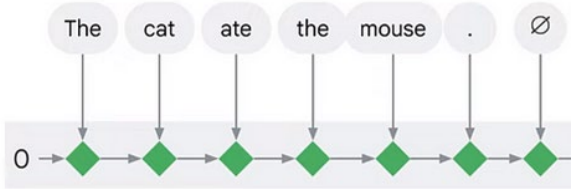
Recap of CNNs

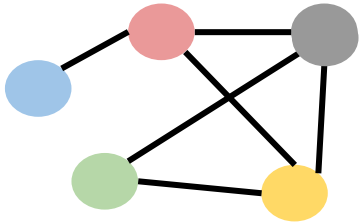
- Excel at processing grid-structured data (images).
- Learn hierarchical features through convolutional filters.
- Achieve state-of-the-art performance in image recognition, object detection, etc.

Geometric Deep Learning

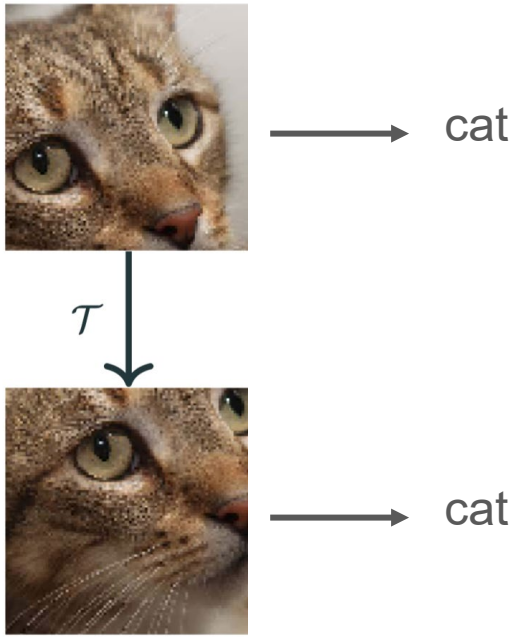
- **Definition:** Geometric Deep Learning refers to techniques that extend deep learning models to non-Euclidean domains, such as graphs, manifolds, and other irregularly structured data.
- **Traditional Deep Learning:** Operates on regular grids like images (2D) or sequences (1D).
- **Geometric Deep Learning:** Generalizes deep learning methods to handle structured data where traditional grids don't apply (e.g., graphs, meshes, and point clouds).

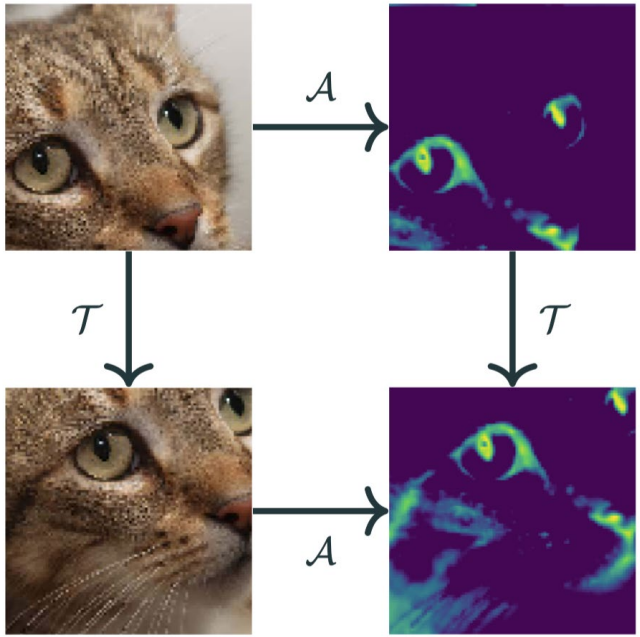






Invariance and Equivariance





https://towardsdatascience.com/a-brief-introduction-to-geometric-deep-learning-dae114923ddb

Invariance and Equivariance in Molecular Modeling

- Molecular properties are invariant to most transformations (e.g., rotation, translation)
  - Solubility
  - Affinity to target
  - ADME kinetics

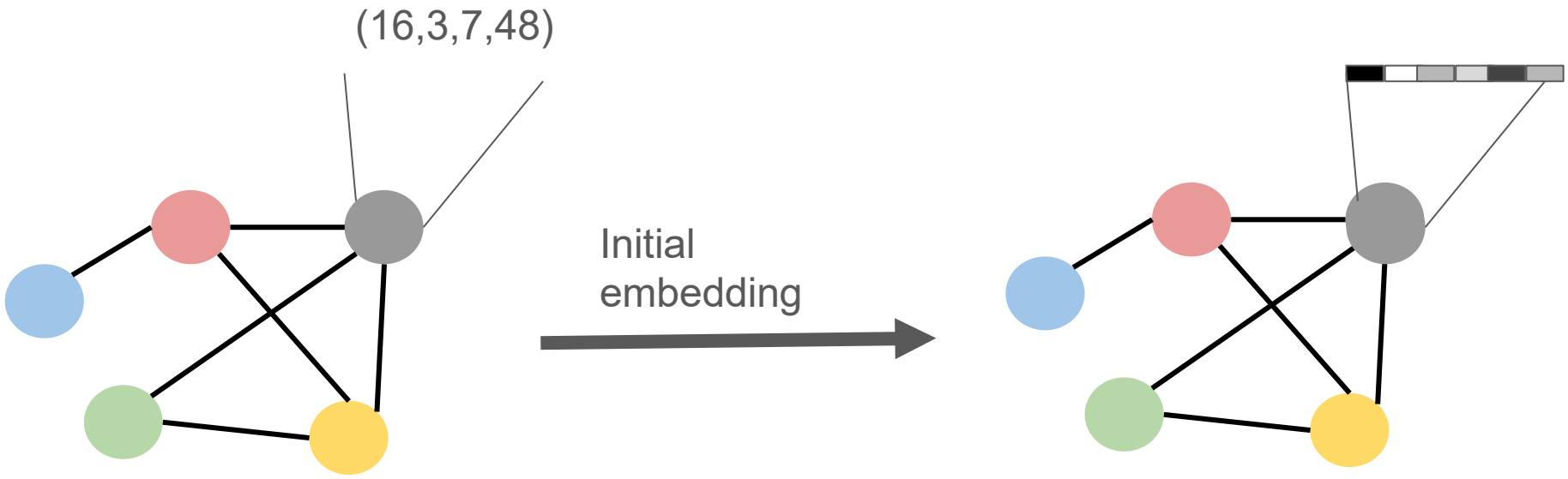
Overview of Graph Neural Networks (GNNs)

- **Definition:** Graph Neural Networks (GNNs) are a class of deep learning models that operate directly on graphs. GNNs are designed to capture the complex relationships and structures between the nodes (e.g., atoms) and edges (e.g., bonds) of a graph.
- **Key Concept:** GNNs learn how to represent graph-structured data (e.g., molecules, social networks) and make predictions based on the relationships between nodes and edges.
- **Applications:** Widely used in cheminformatics, bioinformatics, and social network analysis.

Example of GNN in Drug Discovery

- **Molecular Representation:** A molecule is represented as a graph with atoms as nodes and bonds as edges.
- **Task:** Predict molecular properties (e.g., toxicity, solubility, bioactivity).
- **GNN Processing:** The GNN learns how different atoms (nodes) interact with their neighbors (through bonds/edges) to predict molecular properties based on their structure.
- **Output:** After processing the molecular graph, the GNN outputs a prediction, such as binding affinity, ADME properties, or drug-likeness.

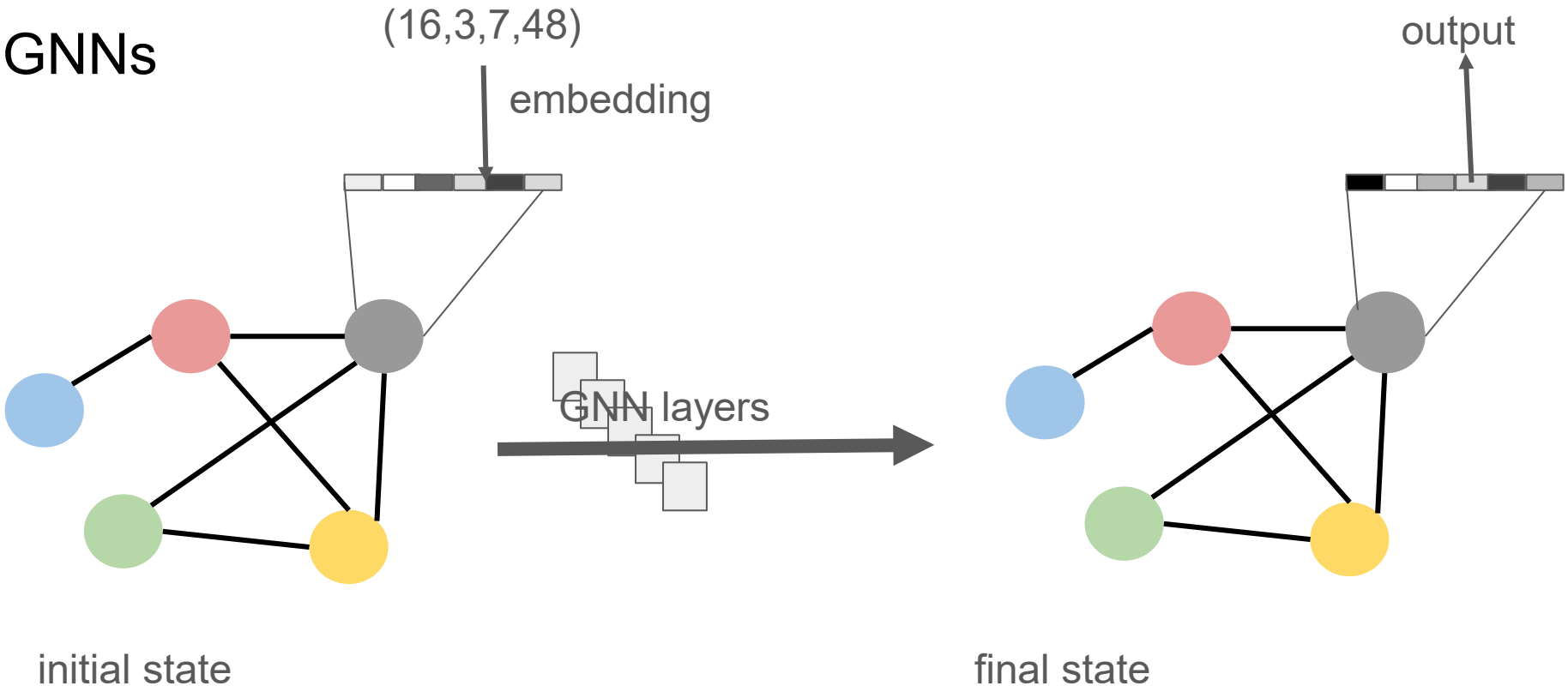
GNNs



graph problem

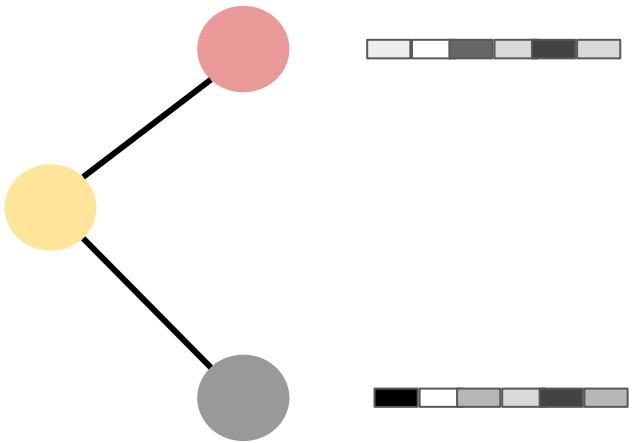
initial representation

GNNs

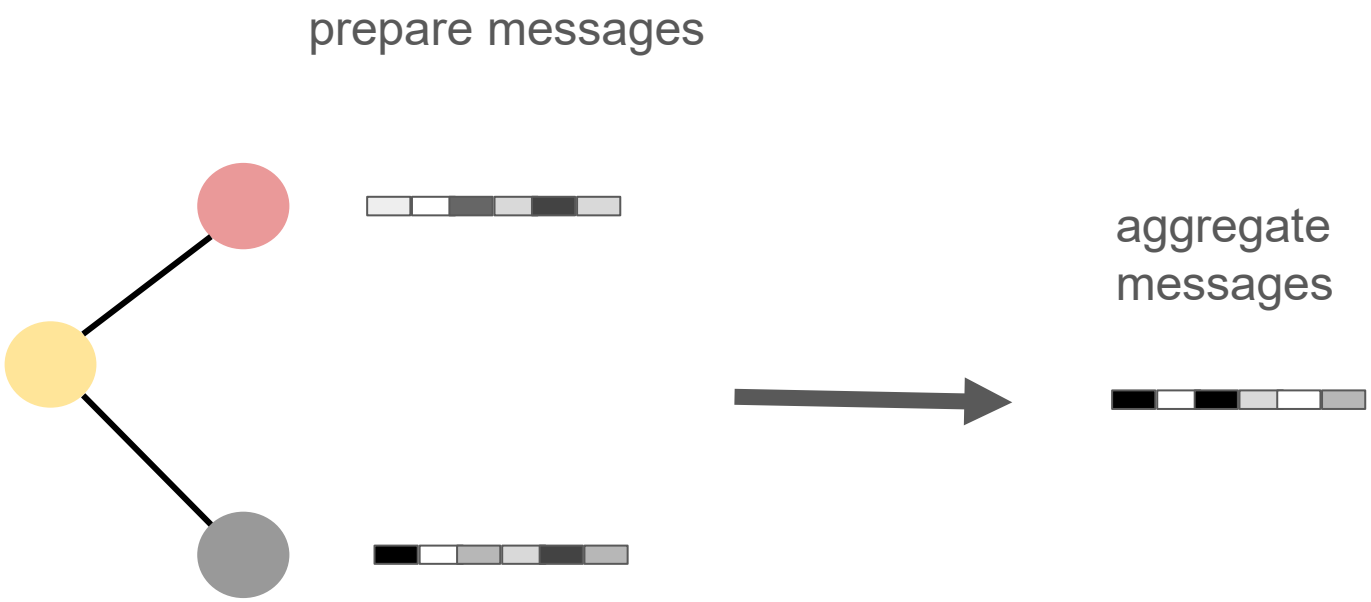

(16,3,7,48)
embedding
output
GNN layers
initial state
final state

GNN layer - message passing

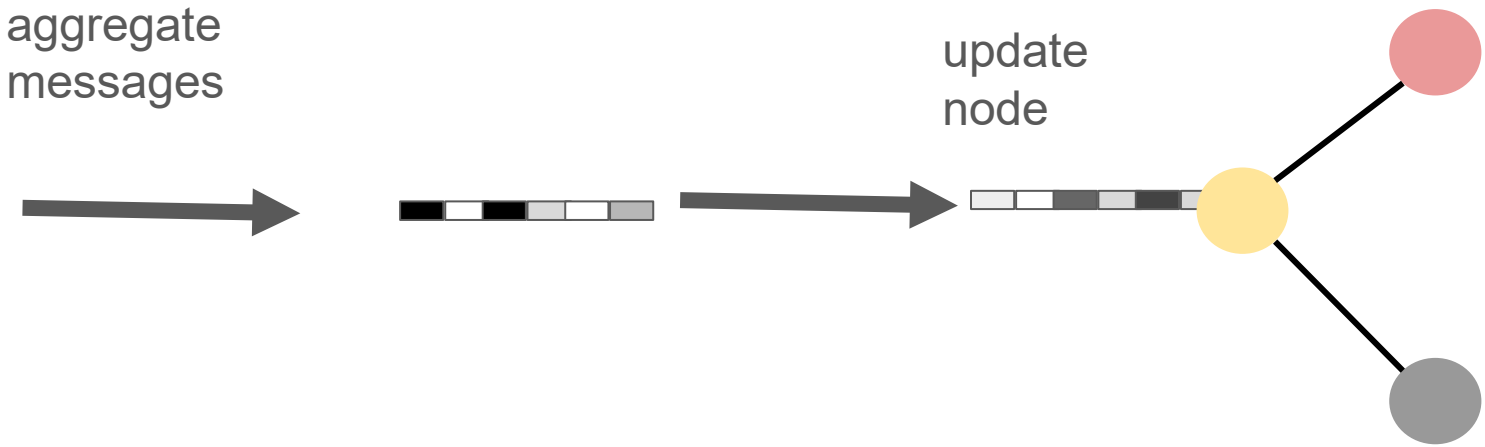
prepare messages



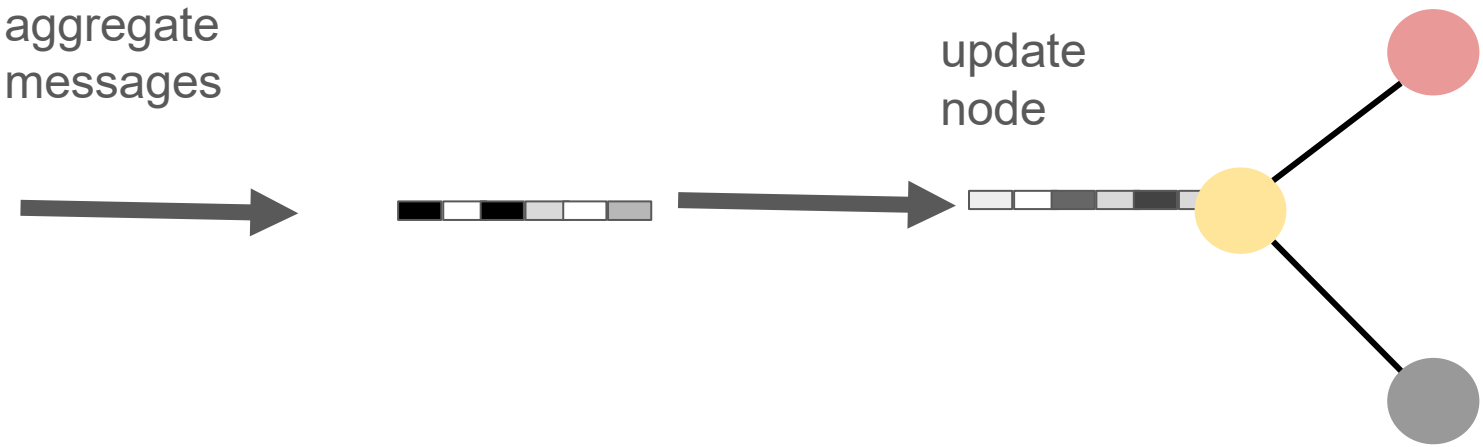
GNN layer - message passing



GNN layer - message passing



GNN layer - message passing



Repeat for each node

Updates in a Basic GNN

$$h_u^{(k)} = \sigma\left(W_{\text{self}}^{(k)} h_u^{(k-1)} + W_{\text{neigh}}^{(k)} \sum_{v \in N_u} h_v^{(k-1)} + b^{(k)} \right)$$

- $h_u^{(k-1)} \in \mathbb{R}^{d^{(k-1)}}$: Node embeddings
- $W_{\text{self}}^{(k)}, W_{\text{neigh}}^{(k)} \in \mathbb{R}^{d^{(k)} \times d^{(k-1)}}$: Learnable parameters
- $b^{(k)} \in \mathbb{R}^{d^{(k)}}$: Bias term
- σ: Elementwise non-linearity (e.g., a tanh or ReLU)

https://deeplearning.cs.cmu.edu/S21/document/slides/11785%20Lecture%20Graph%20Neural%20Networks.pdf

Updates in a Basic GNN

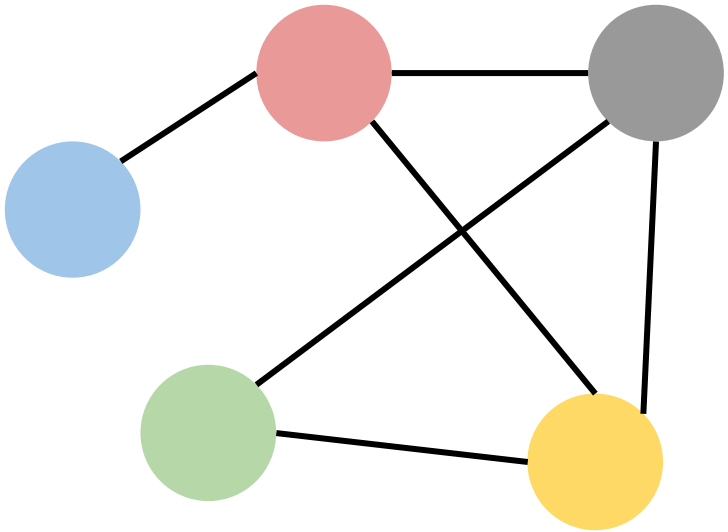
permutation invariant

$$h_u^{(k)} = \sigma\left(W_{\text{self}}^{(k)} h_u^{(k-1)} + W_{\text{neigh}}^{(k)} \sum_{v \in N_u} h_v^{(k-1)} + b^{(k)} \right)$$

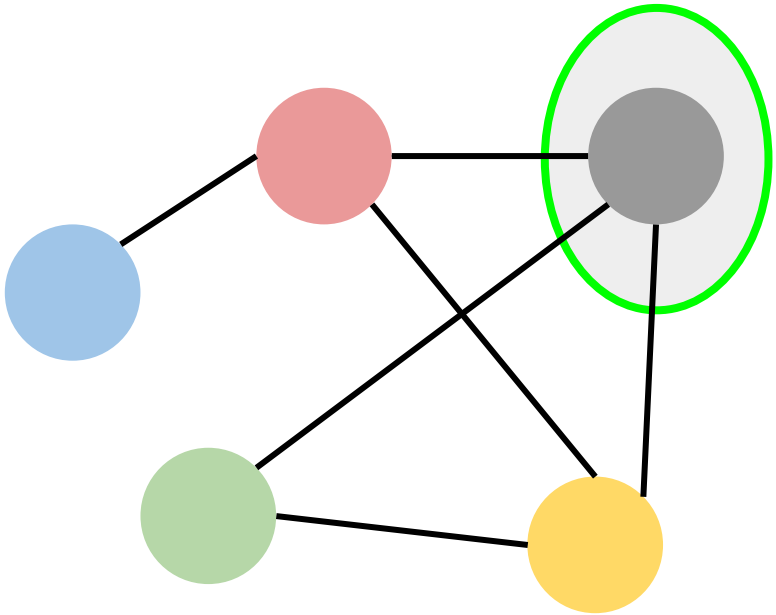
- $h_u^{(k-1)} \in \mathbb{R}^{d^{(k-1)}}$: Node embeddings
- $W_{\text{self}}^{(k)}, W_{\text{neigh}}^{(k)} \in \mathbb{R}^{d^{(k)} \times d^{(k-1)}}$: Learnable parameters
- $b^{(k)} \in \mathbb{R}^{d^{(k)}}$: Bias term
- σ: Elementwise non-linearity (e.g., a tanh or ReLU)

https://deeplearning.cs.cmu.edu/S21/document/slides/11785%20Lecture%20Graph%20Neural%20Networks.pdf

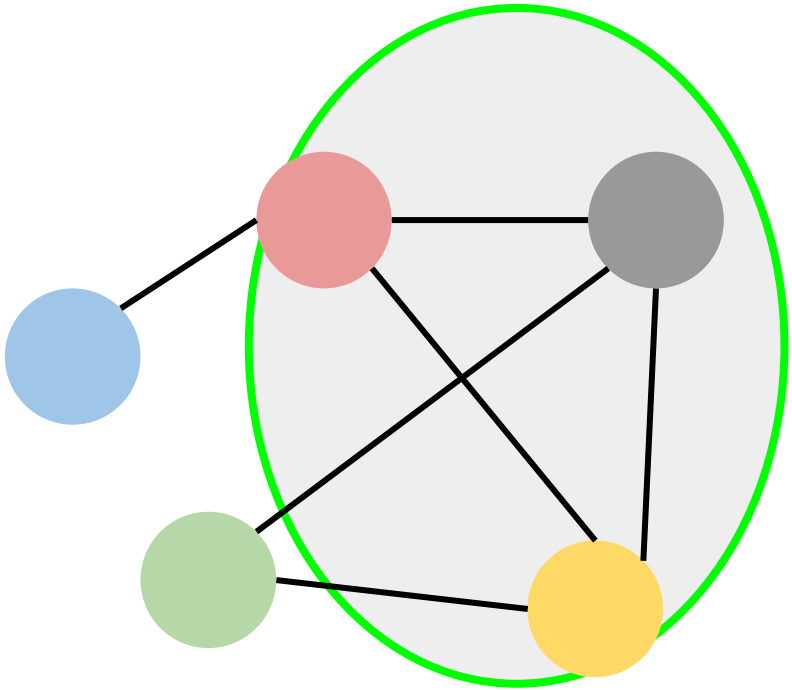
Receptive Field



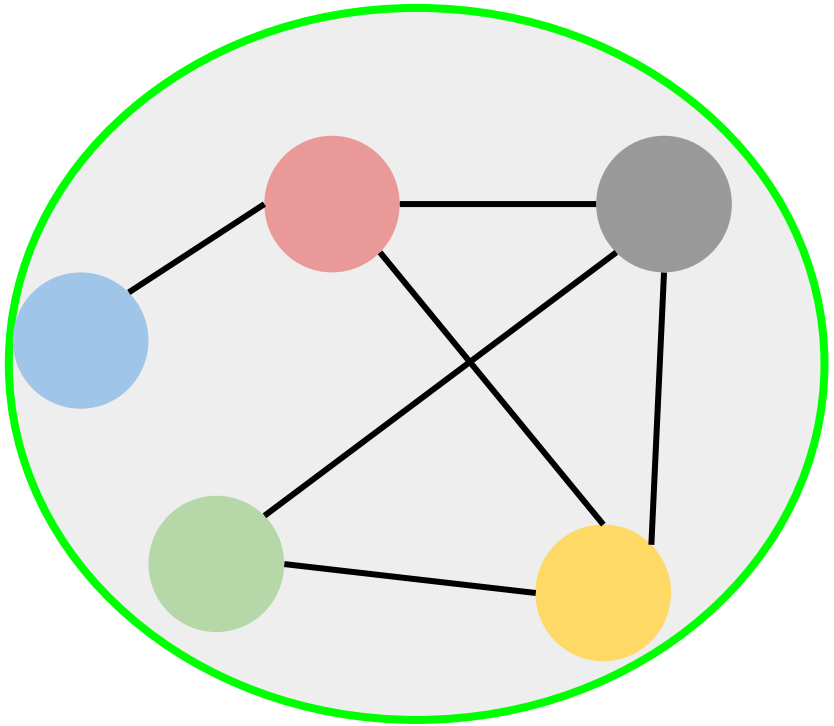
Receptive Field



Receptive Field

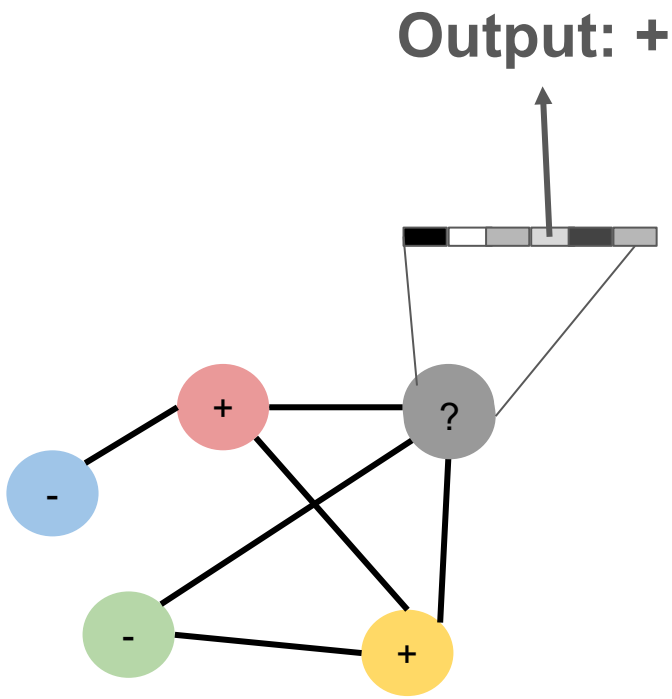


Receptive Field



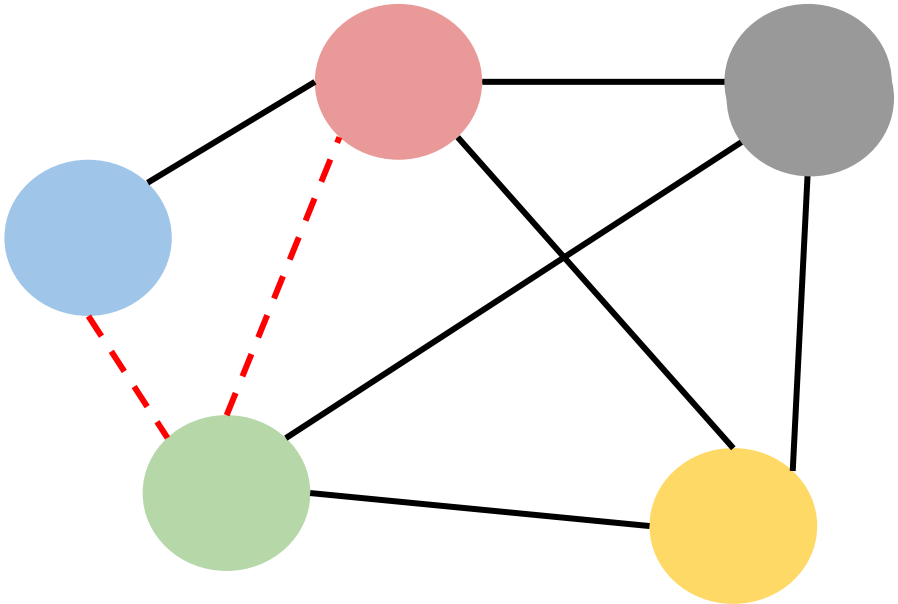
GNN Output - Node Classification

- **Goal:** Predict the label or class of individual nodes in a graph.
- **Input:** Graph with labeled and unlabeled nodes.
- **Output:** Predicted label for each node.
- **Example:** Classifying users in a social network (e.g., spam vs. non-spam users).



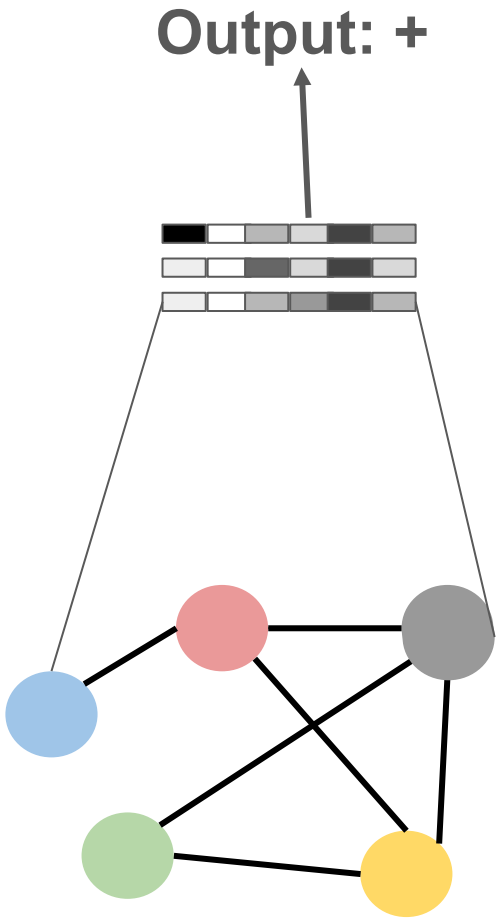
GNN Output - Link Prediction

- **Goal:** Predict the existence of a link (edge) between two nodes.
- **Input:** A graph with some edges missing.
- **Output:** Probability of a link between two nodes.
- **Example:** Recommending friend connections in a social network or predicting drug-target interactions.
- **Approach:**
  - Similarity metric (e.g., cosine similarity)
  - Binary (or other) classifier (e.g., MLP)



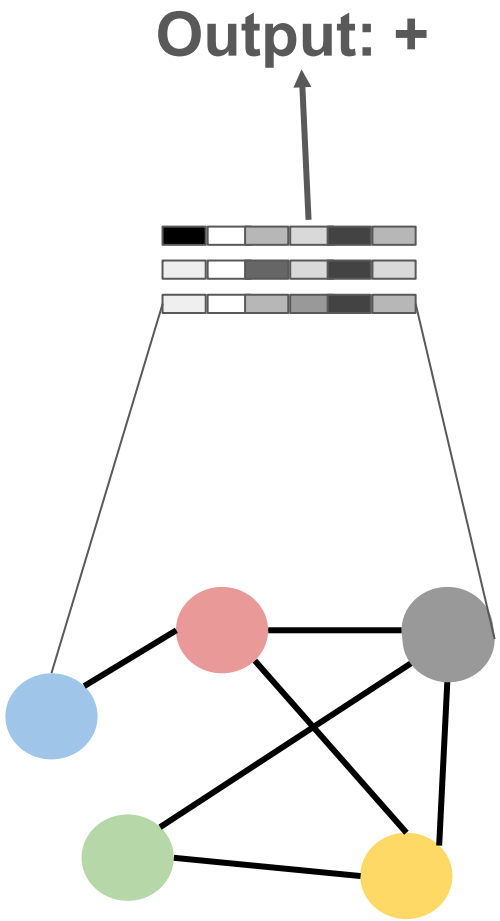
GNN Output - Graph Level

- **Goal:** Predict the label or class of individual nodes in a graph.
- **Input:** Graph with labeled and unlabeled nodes.
- **Output:** Predicted label/value for each node.
- **Example:** Predict binding/solubility/toxicity for molecule



GNN Output - Graph Level

- **Combine node level features:**
  - Global Sum Pooling
  - Global Mean Pooling
  - Global Max Pooling
  - Attention-based Pooling
- **Classification/Regression task:**
  Densely connected layers/MLP



GNNs