Unsupervised Alignment of Hypergraphs with Different Scales

Manh Tuan Do, Kijung Shin
{manh.it97, kijungs}@kaist.ac.kr

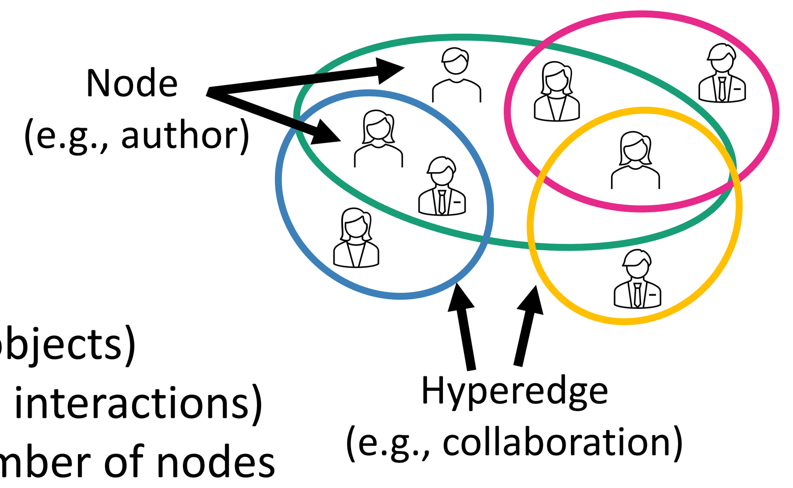
Code & Data: *https://github.com/manhtuando97/HyperAlign*

Summary

- **Novel Problem:** Alignment of hypergraphs in an unsupervised setting
- **Proposed Method: HYPERALIGN** – an alignment method based on three novel components: *structural feature extraction*, *contrastive learning as a pseudo task*, and *topological augmentation* to resolve scale disparity
- **Extensive Experiments:**
  - **Superiority: HYPERALIGN** consistently outperforms eight competitors on twelve real-world hypergraphs in alignment prediction accuracy
  - **Ablation**: each component of **HYPERALIGN** contributes to its performance

Basic Concepts

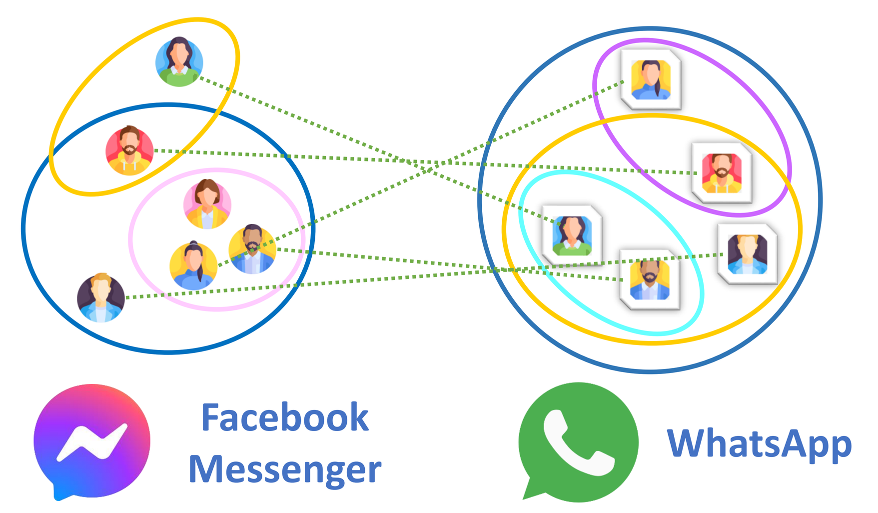
- **Group Interactions are Everywhere:**
  - Co-authors of a research paper
  - Sender & recipients of an email
  - Participants of an online Q/A Session

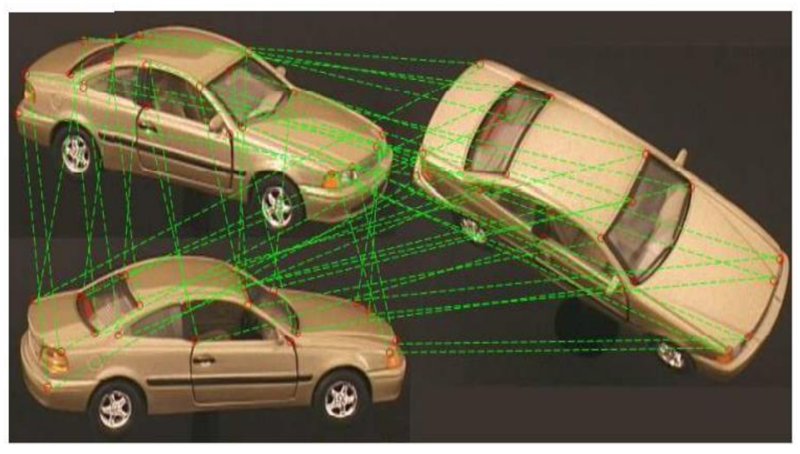


- **Hypergraph $G = (V, E)$:**
  - V: set of nodes (representing people or objects)
  - E: set of hyperedges (representing group interactions)
  - Each *hyperedge* contains an arbitrary number of nodes
- **Hypergraph Alignment (Our Focus):**
  - The problem of identifying the "same nodes" in two given hypergraphs

Application Scenarios

- **User Matching in Social Messaging Platforms (e.g., WhatsApp):**
  - Goal: to identify the same users in different platforms
  - Hypergraph: group chats (hyperedges) among users (nodes)
  - Applications: cross-platform marketing, social behavior analysis, & cybersecurity
- **Object Matching in Images:**
  - Goal: to match features (or pixels) corresponding to the same objects
  - Hypergraph: similar groups (e.g., in terms of colors) of features (or pixels)
  - Applications: medical imaging, image reconstruction, & surveillance



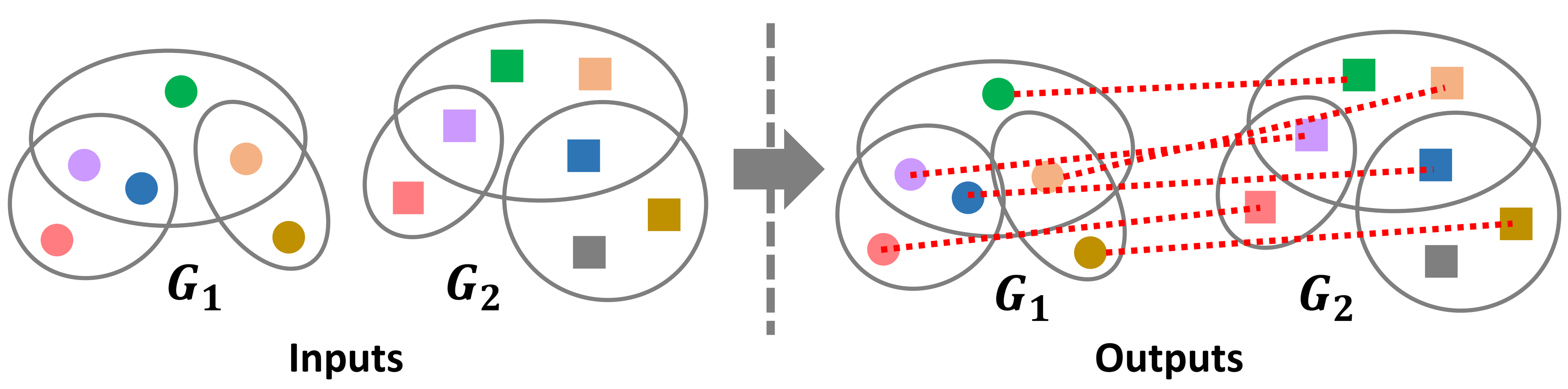


Yan et al., "*Discrete Hyper-Graph Matching*", IEEE CVPR 2015.

Problem Definition

[Unsupervised Alignment of Hypergraphs]

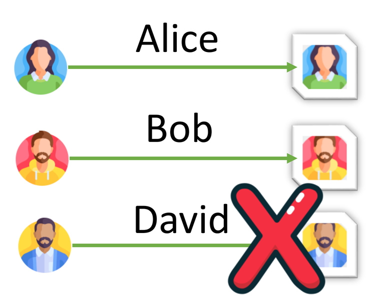
- **Given:** two hypergraphs G_1 and G_2,
- **Find:** the alignment of nodes (or node correspondences) across G_1 and G_2
- **Objective:** to correctly identifies nodes with the same identity
  - No (partial) ground-truth alignment is given
  - No node attributes are given



Challenges

- **Challenge 1: Absence of Node Attributes**
  - Privacy-protection regulations may prevent disclosure of the information of nodes (users)
  - We must rely only on the hypergraph topology

- **Challenge 2: Unsupervised Setting**
  - No (partial) ground-truth alignment is available to guide the alignment of the remaining nodes
  - We must infer correspondences for all nodes



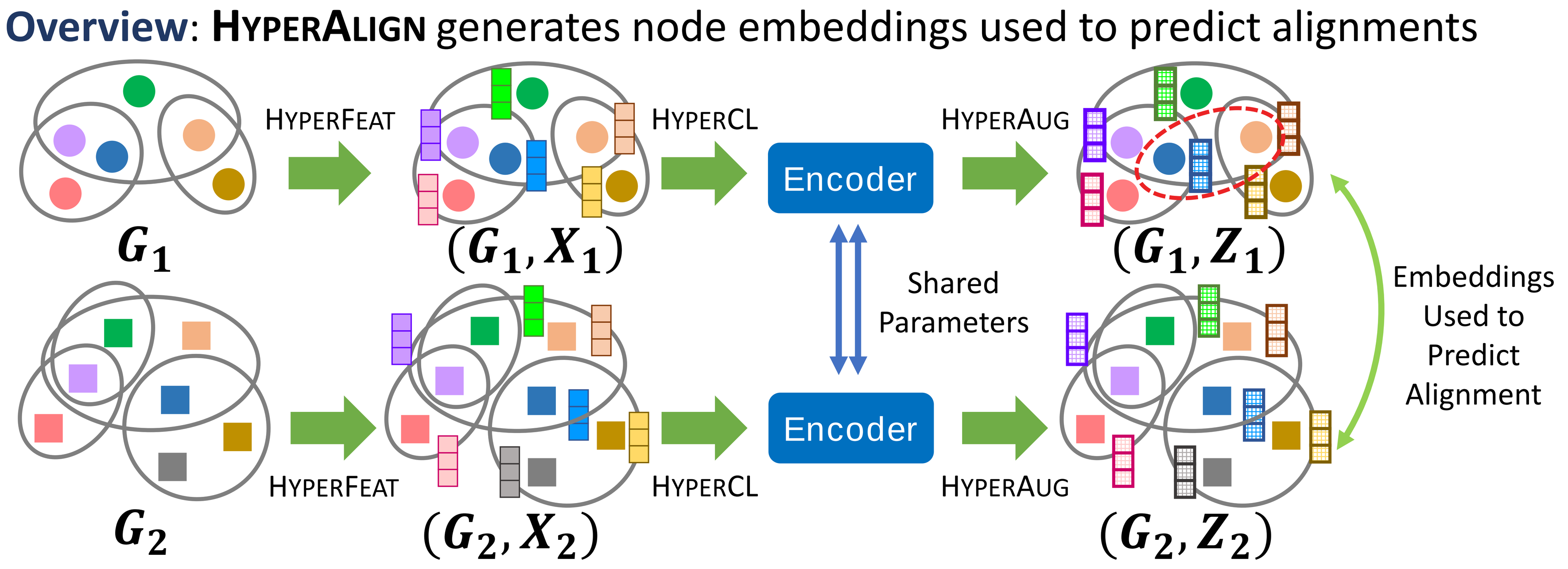
- **Challenge 3: Scale Disparity**
  - Two hypergraphs might be substantially different in sizes
  - For example, one social messaging platform (hypergraph) may have much more group interactions (hyperedges) than the other

Facebook Messenger **(1 billon users)**

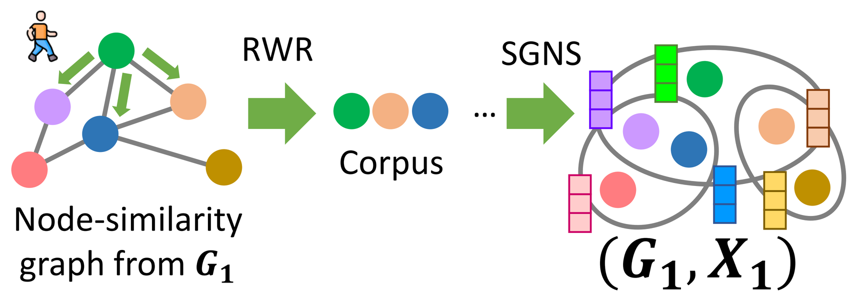
WhatsApp **(2 billion users)**

Proposed Method: HYPERALIGN

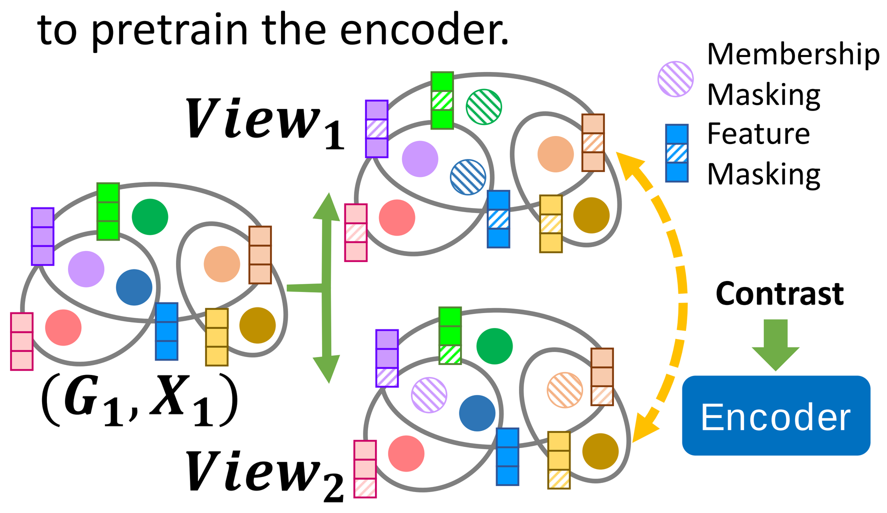
Overview: **HYPERALIGN** generates node embeddings used to predict alignments



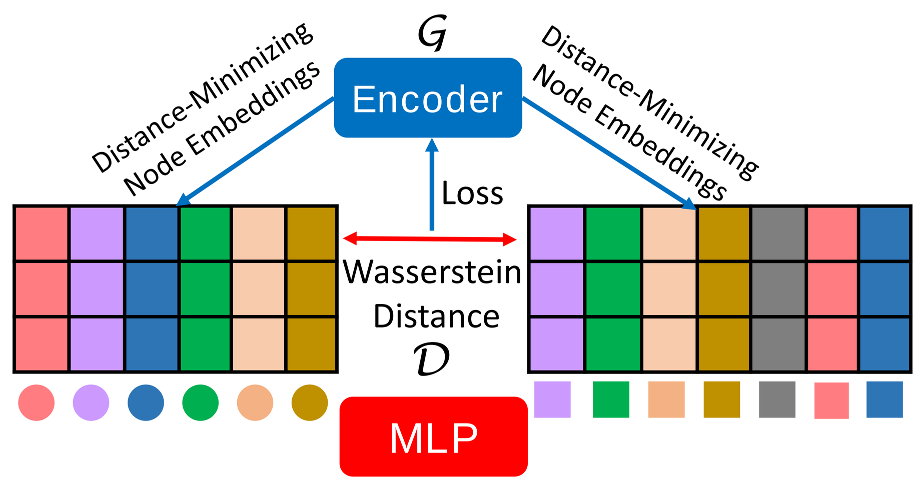
- **Step 1. HYPERFEAT:** extract node features from the hypergraph topology
  - **Build a Corpus:** random walk w/ restart (RWR) on node-similarity graphs
  - **Learn Features:** skip-gram w/ negative sampling (SGNS)



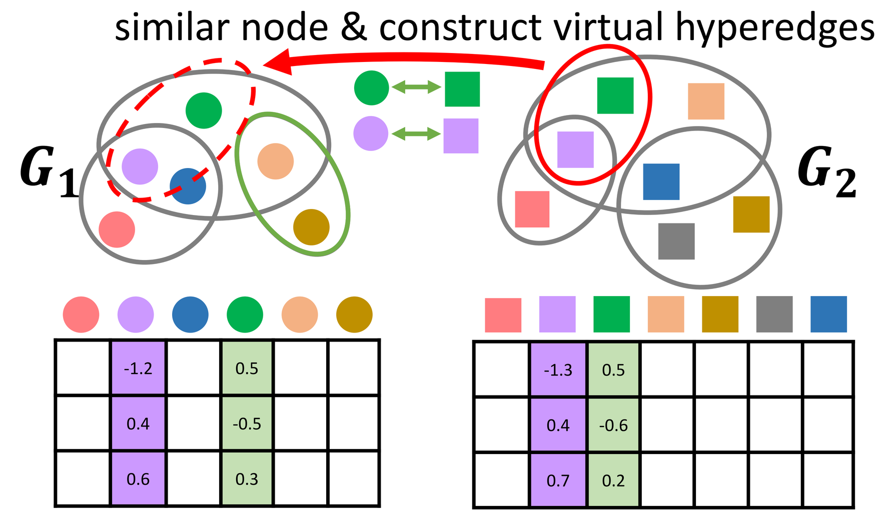
- **Step 2. HYPERCL:** contrastive learning between two views from each hypergraph to pretrain the encoder.



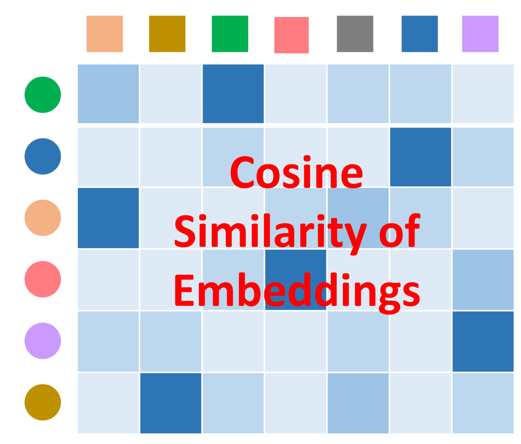
- **Step 3. HYPERAUG:** adversarial learning framework w/ topological augmentation.
  - **Adversarial Learning:** encoder as generator $\mathcal{G}$ & a MLP as Wasserstein Discriminator $\mathcal{D}$



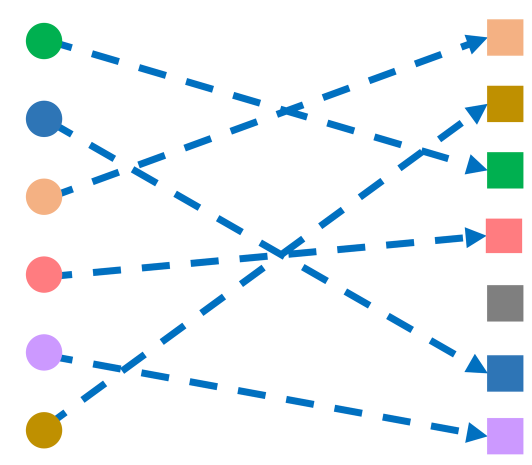
  - **Augmentation:** For each node, find the most similar node & construct virtual hyperedges



- **Step 4. Prediction:** finding the matched node of the most similar embeddings
  - Compute cosine similarities of node embeddings across the two hypergraphs

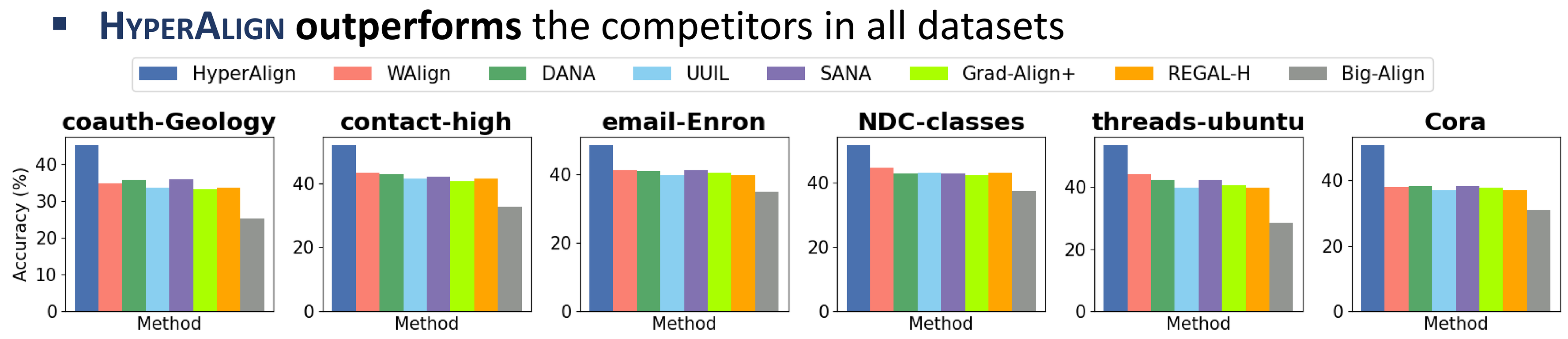


  - Predict each node to correspond with the most similar counterpart node

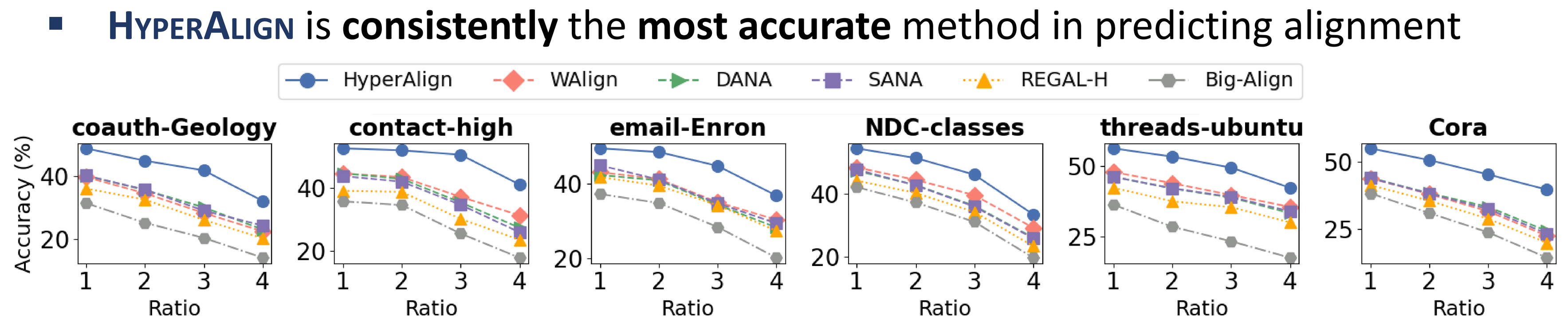


Experiments

- **HYPERALIGN is implementation in** [Python logo]
- **Code and Datasets:**
- **Competitors:** extensions of (bipartite) graph alignment methods
- **EXP 1. Performance:**
  - We compare HYPERALIGN with the competitors in terms of alignment accuracy
  - **HYPERALIGN outperforms** the competitors in all datasets



- **EXP 2. Scale Disparity Ratio of Two Input Hypergraphs:**
  - We vary the scale difference ratio between the two input hypergraphs
  - **HYPERALIGN** is **consistently** the **most accurate** method in predicting alignment



- **EXP 3. Ablation Studies:**
  - We create variants of **HYPERALIGN** by removing/simplifying one or more its modules
  - The full-fledged version **significantly outperforms** the simplified variants, indicating the contribution of each key module of **HYPERALIGN**