One Router to Route Them All: Homogeneous Expert Routing for Heterogeneous Graph Transformers

Georgiy Shakirov
ITMO University

Albert Arakelov
Saint Petersburg State University

Abstract

A common practice in heterogeneous graph neural networks (HGNNs) is to condition model parameters on node and edge types, often under the implicit assumption that type labels provide reliable proxies for underlying semantic roles. However, this approach can lead to overreliance on surface-level type identities and hinder knowledge transfer between structurally or functionally similar entities of different types. In this work, we investigate the integration of Mixture-of-Experts (MoE) architectures into heterogeneous graph learning—a direction that remains underexplored despite the success of MoE in homogeneous domains. Within this broader investigation, we specifically challenge the necessity of type-specific expert specialization in MoE-based HGNNs. We hypothesize that a *shared expert pool*, combined with a routing mechanism that is partially invariant to node-type information, encourages experts to specialize on latent semantic patterns rather than syntactic type labels—thereby enabling robust cross-type generalization. To test this hypothesis, we propose *Homogeneous Expert Routing* (HER), an integrated MoE layer atop the Heterogeneous Graph Transformer (HGT), which stochastically masks type embeddings during training to regularize the router against type-dependent shortcuts. Evaluated on multiple heterogeneous graph benchmarks—including IMDB, ACM, and DBLP—HER consistently outperforms both standard HGT and a type-separated MoE baseline in link prediction. In-depth analysis on IMDB reveals that HER experts develop type-agnostic, semantically coherent specializations, such as alignment with movie genres, confirming that routing is driven by shared semantics rather than type identity. Our results establish that regularizing the influence of type information in expert routing leads to more generalizable, efficient, and interpretable representations, offering a new design principle for heterogeneous graph learning.

1 Introduction

Heterogeneous graphs—comprising multiple node and edge types—are a natural abstraction for complex relational systems, from knowledge graphs [Wang et al., 2019] to recommendation platforms [Fan et al., 2019]. State-of-the-art heterogeneous GNNs, such as the Heterogeneous Graph Transformer (HGT) [Hu et al., 2020], employ type-aware message passing to capture inter-type dependencies. While HGT enables rich cross-type contextualization through its attention mechanism, its reliance on type-specific parameters—and the potential for downstream components to condition processing directly on explicit type labels—can still lead to overreliance on syntactic identity rather than underlying semantic content.

Recent advances in Mixture-of-Experts (MoE) architectures [Shazeer et al., 2017, Fedus et al., 2022b] demonstrate that sparse, dynamic routing to specialized subnetworks can scale model capacity while maintaining computational efficiency. Crucially, in their original domains (e.g., language and vision), MoE operates on homogeneous inputs—there are no explicit categorical labels guiding expert selection, and specialization emerges purely from data-driven routing.

In the graph domain, Graph Mixture of Experts (GMoE) [Zhang et al., 2023] successfully adapts MoE to large-scale *homogeneous* graphs, improving scalability and inducing functional diversity. However, GMoE does not address heterogeneity, nor does it consider how node-type metadata should interact with routing decisions.

However, in heterogeneous graphs, a naive design choice would be to assign *disjoint expert sets per node type*—a strategy we refer to as *SeparatedMoE*. While this ensures type-specific processing, it fundamentally prevents knowledge transfer at the expert level: even if HGT has already fused information

across types into a node's representation, the subsequent MoE layer processes actors, directors, and movies through entirely separate expert pools. Consequently, semantically similar patterns—e.g., "frequent collaborators" among actors and directors—must be redundantly modeled by unrelated experts, limiting parameter efficiency and generalization.

One might consider sharing a single expert pool across all types to enable cross-type alignment. However, if the router has unrestricted access to explicit type embeddings, it can trivially learn deterministic "type $\rightarrow$ expert" mappings, effectively collapsing back to type-separated behavior despite shared parameters. To address this, we propose a controlled use of type information: we *do* provide type embeddings to the router to preserve its ability to distinguish structural roles when beneficial, but we *regularize* this dependency during training via stochastic masking—randomly zeroing out type embeddings with probability p_{mask}. This forces the router to rely increasingly on the semantic content of HGT-processed node states when type cues are withheld, encouraging experts to specialize on latent, type-agnostic patterns.

We formalize this approach in *Homogeneous Expert Routing* (Definition 3) —a novel MoE framework for heterogeneous graphs that combines a shared expert pool with a homogenizing router regularized by stochastic type masking. Evaluated on multiple standard benchmarks (IMDB, ACM, DBLP), HER consistently outperforms both standard HGT and SeparatedMoE in link prediction. Moreover, qualitative analysis on IMDB reveals that HER experts spontaneously align with interpretable semantic concepts (e.g., movie genres), and nodes of different types co-cluster under the same experts—demonstrating genuine cross-type semantic specialization.

Our contributions are threefold:

- We identify a fundamental limitation of type-separated MoE in heterogeneous graphs: despite HGT's ability to fuse information across types, disjoint expert pools prevent knowledge transfer at the expert level, forcing redundant modeling of semantically similar patterns (e.g., "frequent collaborators" among actors and directors).
- We propose *Homogeneous Expert Routing* (HER)—a novel MoE framework that employs stochastic masking of type embeddings during routing to explicitly regularize against type-dependent shortcuts, thereby encouraging a shared expert pool to specialize on latent, type-agnostic semantic roles rather than syntactic labels.
- We validate HER on multiple heterogeneous benchmarks (IMDB, ACM, DBLP), demonstrating consistent performance gains over both standard HGT and SeparatedMoE in link prediction, and provide quantitative evidence of semantic specialization—including cross-type co-clustering and alignment between expert assignments and external semantic labels (e.g., movie genres).

This work bridges two active research directions—heterogeneous GNNs and graph MoE—and offers a principled approach to expert routing, demonstrating that superior performance and interpretability in heterogeneous graph learning can be achieved by decoupling expert specialization from node type, allowing semantic roles to emerge from data while using type information only as a regularized, non-deterministic cue.

2 Related Work

2.1 Heterogeneous Graph Neural Networks

Modeling heterogeneous graphs has been a central challenge in graph representation learning. Early approaches, such as Metapath2Vec [Dong et al., 2017], relied on handcrafted meta-paths to capture semantic relations. More recent methods adopt end-to-end neural architectures that learn type-aware representations without explicit path engineering. Among these, the Heterogeneous Graph Transformer (HGT) [Hu et al., 2020] stands out for its use of type-dependent attention mechanisms and dynamic parameter generation, enabling scalable and expressive message passing across diverse node and edge types. HGT has become a de-facto backbone for heterogeneous graph learning and is the foundation of our architecture. However, like most HGNNs, HGT ties transformation logic directly to node types, which can impede generalization when type distributions shift or when rare types appear at test time–a limitation our work explicitly addresses.

Other notable HGNNs include RGCN [Schlichtkrull et al., 2018], which uses relation-specific weight matrices, and HetGNN [Zhang et al., 2019], which employs type-specific aggregators with Bi-LSTMs. While effective, these models similarly assume that node types provide sufficient inductive bias for specialization, an assumption we relax through semantic-driven expert routing.

2.2 Mixture-of-Experts in Deep Learning

The Mixture-of-Experts (MoE) paradigm, introduced by Jacobs et al. [1991] and revitalized by Shazeer et al. [2017], decomposes a large model into sparsely activated subnetworks ("experts") selected by a trainable "router." This design improves model capacity while controlling computational cost, and has been successfully applied in vision [Puigcerver et al., 2021], language [Fedus et al., 2022b], and multimodal learning. A key challenge in MoE is ensuring expert diversity and load balancing, typically addressed via auxiliary losses that encourage uniform expert utilization [Fedus et al., 2022a].

In the graph domain, Graph Mixture of Experts (GMoE) [Zhang et al., 2023] adapts MoE to large-scale homogeneous graphs by routing nodes to experts based on structural and feature similarity. GMoE demonstrates improved scalability and representation diversity but does not consider node or edge types, and its routing mechanism assumes a single semantic space. Consequently, GMoE is not designed for–and cannot be trivially extended to–heterogeneous settings where type metadata plays a critical role. Our work fills this gap by rethinking MoE routing in the presence of type information, proposing that type should inform feature representation but not dictate routing decisions.

2.3 Type-Agnostic and Semantic-Aware Graph Learning

A growing line of work questions the necessity of explicit type conditioning. UniGNN [Huang et al., 2022] proposes unified message passing functions across types to reduce parameter overhead, while SimpleHGN [Lv et al., 2021] uses learnable type embeddings without type-specific transformations. However, these methods still use type embeddings as direct inputs to the main network, potentially allowing the model to shortcut through type identity rather than learning deeper semantics.

More closely related to our motivation, semantic role learning has been explored in knowledge graphs [Wang et al., 2021] and social networks [Liu et al., 2022], where the goal is to discover functional roles (e.g., "influencer", "bridge") that transcend categorical labels. Our Homogeneous Expert Router can be viewed as a mechanism for inducing semantic roles via routing: by masking type during expert selection, we force the model to discover such roles implicitly from data.

While semantic role learning has been studied in other contexts, the integration of Mixture-of-Experts with heterogeneous graph learning–and specifically the role of node-type information in expert routing–has not been systematically investigated. Our work is the first to propose and empirically validate a routing mechanism that deliberately limits access to type identity to promote cross-type semantic specialization.

3 Method

3.1 Preliminaries

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a heterogeneous graph, where $\mathcal{V}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of directed edges. Each node $v \in \mathcal{V}$ has a type given by $\tau(v) \in \mathcal{T}$, and each edge $e = (u, v) \in \mathcal{E}$ has a type given by $\phi(e) \in \Phi$. For clarity, we denote an edge from node u to node v of edge type $\phi(e)$ as the triplet (u, e, v).

Each node v is associated with an input feature vector $\mathbf{x}_v \in \mathbb{R}^{d_{\tau(v)}}$, where the input dimension $d_{\tau(v)}$ may vary by node type. After an initial type-specific linear projection (implicit in the first HGT layer), all node representations reside in a shared hidden space $\mathbb{R}^d$.

We build upon the Heterogeneous Graph Transformer (HGT) [Hu et al., 2020]. Let $\mathbf{H}^{(\ell-1)} \in \mathbb{R}^{|\mathcal{V}| \times d}$ denote the matrix of node embeddings at layer $\ell - 1$, with $\mathbf{h}_v^{(\ell-1)}$ being the embedding of node v. For each edge $(u, e, v) \in \mathcal{E}$, HGT computes attention and messages as follows.

Multi-head attention for edge (u, e, v): The attention head i on edge (u, e, v) is:

$$\mathbf{a}^{(i)}_{(u,e,v)} = \mathbf{k}^{(i)}_u \mathbf{W}^{\text{att}}_{\phi(e)} \left(\mathbf{q}^{(i)}_v\right)^\top \cdot \frac{\mu_{(\tau(u),\phi(e),\tau(v))}}{\sqrt{d}}, \tag{1}$$

$$\mathbf{a}^{(i)}_{(u,e,v)} = \left(\mathbf{k}^{(i)}_u\right)^\top \mathbf{W}^{\text{att}}_{\phi(e)} \mathbf{q}^{(i)}_v \cdot \frac{\mu_{(\tau(u),\phi(e),\tau(v))}}{\sqrt{d/h}}, \tag{2}$$

where

$$\mathbf{k}^{(i)}_u = \mathbf{W}^{K,(i)}_{\tau(u)} \mathbf{h}^{(\ell-1)}_u,$$
$$\mathbf{q}^{(i)}_v = \mathbf{W}^{Q,(i)}_{\tau(v)} \mathbf{h}^{(\ell-1)}_v,$$

$\mathbf{W}^{K,(i)}_{\tau(u)}, \mathbf{W}^{Q,(i)}_{\tau(v)} \in \mathbb{R}^{d \times \frac{d}{h}}$ are learnable projection matrices specific to node types and attention head i. The $\mu_{(\tau(u),\phi(e),\tau(v))} \in \mathbb{R}^{|\mathcal{T}| \times |\Phi| \times |\mathcal{T}|}$ to denote the general significance of each meta relation triplet, serving as an adaptive scaling to the attention and $\mathbf{W}^{\text{att}}_{\phi(e)} \in \mathbb{R}^{\frac{d}{h} \times \frac{d}{h}}$ is a learnable matrix specific to edge type $\phi(e)$.

The h-head attention for each edge $e = (u, v)$ is:

$$\mathbf{attention}_{(u,e,v)} = \underset{\forall u \in \mathcal{N}(v)}{\text{Softmax}} \left(\Big\|_{i=1}^{h} \mathbf{a}^{(i)}_{(u,e,v)} \right), \tag{3}$$

where $\Big\|$ denotes concatenation across heads (with an implicit linear projection absorbed into the message path).

Message computation for edge (u, e, v): The message from u to v under head i is:

$$\mathbf{m}^{(i)}_{(u,e,v)} = \mathbf{W}^{\text{msg}}_{\phi(e)} \left(\mathbf{W}^{msg,(i)}_{\tau(u)} \mathbf{h}^{(\ell-1)}_u \right), \tag{4}$$

with $\mathbf{W}^{\text{msg},(i)}_{\tau(u)} \in \mathbb{R}^{d \times \frac{d}{h}}$ and $\mathbf{W}^{\text{msg}}_{\phi(e)} \in \mathbb{R}^{\frac{d}{h} \times \frac{d}{h}}$. The full message is:

$$\mathbf{message}_{(u,e,v)} = \Big\|_{i=1}^{h} \mathbf{m}^{(i)}_{(u,e,v)}. \tag{5}$$

Aggregation and update: For each target node v, messages are aggregated over its incoming edges:

$$\widetilde{\mathbf{h}}^{(\ell)}_v = \sum_{\forall u \in \mathcal{N}(v)} \mathbf{attention}_{(u,e,v)} \cdot \mathbf{message}_{(u,e,v)}, \tag{6}$$

The updated embedding includes a residual connection and a type-specific output transformation:

$$\mathbf{h}^{(\ell)}_v = \mathbf{W}^{A}_{\tau(v)} \, \sigma\left(\widetilde{\mathbf{h}}^{(\ell)}_v\right) + \mathbf{h}^{(\ell-1)}_v, \tag{7}$$

where σ is a non-linear activation (e.g., GELU), and $\mathbf{W}^{A}_{\tau(v)} \in \mathbb{R}^{d \times d}$ is a learnable matrix specific to node type $\tau(v)$.

MoE architectures for Heterogeneous Graphs. We consider two design paradigms for integrating Mixture-of-Experts (MoE) into HGNNs:

Definition 1 (SeparatedMoE)**.** *A SeparatedMoE architecture is defined by the triple $(\mathcal{P}, E, R)$, where:*

1. *$\mathcal{P} = \{E_\tau \subset E \mid \tau \in \mathcal{T}\}$ is a partition of the set E, indexed by the set of types, such that: (a) $\bigcup_\tau E_\tau = E$, (b) $\forall \tau, \kappa \in \mathcal{T}, \tau \neq \kappa : E_\tau \cap E_\kappa = \emptyset$, (c) $\forall \tau \in \mathcal{T} : E_\tau \neq \emptyset$*
2. *$R : \mathcal{V} \times \mathbb{R}^d \to \mathcal{P}(E)$ satisfies the* type-locking condition*:*

$$\forall v \in \mathcal{V}, \forall x \in \mathbb{R}^d : \quad R(v, x) \subseteq E_{\tau(v)}.$$

Definition 2 (SharedMoE)**.** *A* SharedMoE *architecture is defined by the tuple (E, R), where:*

1. $E = \{E_1, \ldots, E_K\}$ *is a* global expert pool*: a single set of K experts that is* ***accessible to all nodes in the graph, regardless of their type****.*
2. $R : \mathcal{V} \times \mathbb{R}^d \to 2^E$ *is a routing function that assigns each node to a subset of E. The architecture* ***imposes no type-based partitioning*** *of the expert pool, meaning that for any two node types $\tau, \tau' \in \mathcal{T}$, experts in E may be selected by nodes of both types. This enables cross-type expert sharing by design.*

When instantiated with Homogeneous Expert Routing *(Definition 3), the routing function R operates on fused representations $(\mathbf{X}, \widetilde{\mathbf{T}})$, where $\widetilde{\mathbf{T}}$ denotes stochastically masked type embeddings, thereby regularizing against deterministic type-to-expert mappings while preserving access to the shared pool E.*

Definition 3 (Homogeneous Expert Routing)**.** *The routing function R maps fused node representations and type embeddings to a matrix of expert assignment probabilities:*

$$R : (\mathbf{X}, \widetilde{\mathbf{T}}) \to \mathbf{G},$$

where $\mathbf{X} \in \mathbb{R}^{N \times d}$ is the matrix of HGT-processed node embeddings, $\widetilde{\mathbf{T}} \in \mathbb{R}^{N \times d}$ contains corresponding (possibly masked) type embeddings, and $\mathbf{G} \in \mathbb{R}^{N \times K}$ is the routing matrix with $G_{v,i} = \Pr(\text{node } v \text{ selects expert } i)$. Under Homogeneous Expert Routing (HER), $\widetilde{\mathbf{T}}$ is stochastically masked during training, making R a regularized, type-robust mapping that prioritizes semantic content in $\mathbf{X}$ for expert selection.

3.2 Shared Mixture-of-Experts with Homogenizing Routing

To induce semantic specialization without type dependency, we introduce a shared MoE layer that processes post-HGT embeddings $\mathbf{H} = [\mathbf{h}_1, ..., \mathbf{h}_N] \in \mathbb{R}^{N \times d}$ (where $N = |\mathcal{V}|$) via a homogenizing router. Let $\mathbf{E} \in \mathbb{R}^{|\mathcal{T}| \times d}$ contain learnable type embeddings, with row v as $\mathbf{e}_{\tau(v)} \in \mathbb{R}^d$. The core idea is to allow the routing function access to type information, but to regularize its reliance on it—thereby preventing deterministic "type → expert" shortcuts and encouraging routing based on latent semantics. As we will formally justify in Section 4, this stochastic masking strategy provably encourages the router to extract semantic signals from the node representation itself, rather than exploiting surface-level type cues.

Input Fusion For each node v, we define a fused representation $\tilde{\mathbf{x}}_v \in \mathbb{R}^d$ via one of three strategies:

$$\tilde{\mathbf{x}}_v = \begin{cases} \mathbf{h}_v & \text{if } \texttt{combine\_type} = \texttt{'none'}, \\ \mathbf{W}_h \mathbf{h}_v + \tilde{\mathbf{e}}_v & \text{if } \texttt{combine\_type} = \texttt{'add'}, \\ \mathbf{W}_h \mathbf{h}_v \odot \tilde{\mathbf{e}}_v & \text{if } \texttt{combine\_type} = \texttt{'mul'}, \end{cases} \tag{8}$$

where $\tilde{\mathbf{e}}_v$ is a stochastically masked version of the type embedding:

$$\tilde{\mathbf{e}}_v = m_v \cdot \mathbf{e}_{\tau(v)}, \quad m_v \sim \text{Bernoulli}(1 - p_{\text{mask}}). \tag{9}$$

Here, $\mathbf{W}_h \in \mathbb{R}^{d \times d}$ is learnable, and $p_{\text{mask}} \in [0, 1]$ controls type masking probability. During inference, we set $m_v = 1$.

Routing The router computes expert gates over the fused inputs $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_N]^\top$:

$$\mathbf{G} = \text{softmax}\left(\widetilde{\mathbf{X}} \mathbf{W}_r\right) \in \mathbb{R}^{N \times K}, \tag{10}$$

with $\mathbf{W}_r \in \mathbb{R}^{d \times E}$. Stochastic masking ensures type-invariance in expectation for $p_{\text{mask}} > 0$. For each node v, select the top-k experts:

$$\mathcal{K}_v = \text{top-}k(\mathbf{G}_{v,:}), \quad \mathbf{g}_v = \text{softmax}\left(\mathbf{G}_{v,\mathcal{K}_v}\right).$$

Expert Processing Each expert $i \in \{1, \ldots, K\}$ is a two-layer feedforward network:

$$\text{Expert}_i(\mathbf{x}) = \mathbf{W}_2^{(i)} \sigma\left(\mathbf{W}_1^{(i)} \mathbf{x} + \mathbf{b}_1^{(i)}\right) + \mathbf{b}_2^{(i)}, \tag{11}$$

with $\mathbf{W}_1^{(i)} \in \mathbb{R}^{d' \times d}$, $\mathbf{W}_2^{(i)} \in \mathbb{R}^{d' \times d'}$, and $\sigma = \text{ReLU}$. All experts share the same input $\tilde{\mathbf{x}}_v$. The final output for node v is:

$$\mathbf{y}_v = \sum_{i \in \mathcal{K}_v} g_{v,i} \cdot \text{Expert}_i(\tilde{\mathbf{x}}_v). \tag{12}$$

Load Balancing Loss To encourage uniform expert usage, we adopt the auxiliary load balancing loss from Fedus et al. [2022a]:

$$\mathcal{L}_{\text{bal}} = K \cdot \sum_{i=1}^{K} f_i P_i, \tag{13}$$

where

$$f_i = \frac{1}{N} \sum_{v=1}^{N} \sum_{j \in \mathcal{K}_v} \mathbb{I}[j = i]\, g_{v,j}, \tag{14}$$

$$P_i = \frac{1}{N} \sum_{v=1}^{N} G_{v,i}, \tag{15}$$

are the fraction of total routing weight and the mean router probability assigned to expert i, respectively. Here, N denotes the number of nodes in the batch (or graph), and E is the total number of experts. Unlike the original Switch Transformer formulation that uses hard routing via arg max, our model employs soft top-k routing; thus, f_i captures the average routing weight directed to expert i, rather than the count of nodes assigned to it. The total training loss combines the task-specific objective with this balancing term:

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \alpha\, \mathcal{L}_{\text{bal}}, \tag{16}$$

where $\alpha > 0$ is a hyperparameter controlling the strength of the load balancing constraint.

3.3 Full Architecture: HGT-SharedMoE

Our model stacks B identical blocks, each consisting of:

- An HGT convolution layer performing type-aware message passing (Section 3.1),
- A shared Mixture-of-Experts (MoE) layer (Section 3.2), applied to all node types.

Crucially, the *same* MoE layer is used across all node types–in contrast to type-separated variants that maintain independent MoE modules per type. This design enforces parameter sharing and enables cross-type knowledge transfer.

The final node embeddings $\{\mathbf{y}_u, \mathbf{y}_v\}$ are used for link prediction via a bilinear decoder:

$$\hat{y}_{uv} = \sigma\left(\mathbf{y}_u^\top \mathbf{M} \mathbf{y}_v\right), \tag{17}$$

where $\mathbf{M} \in \mathbb{R}^{d \times d}$ is a learnable matrix and $\sigma(\cdot)$ is the sigmoid function.

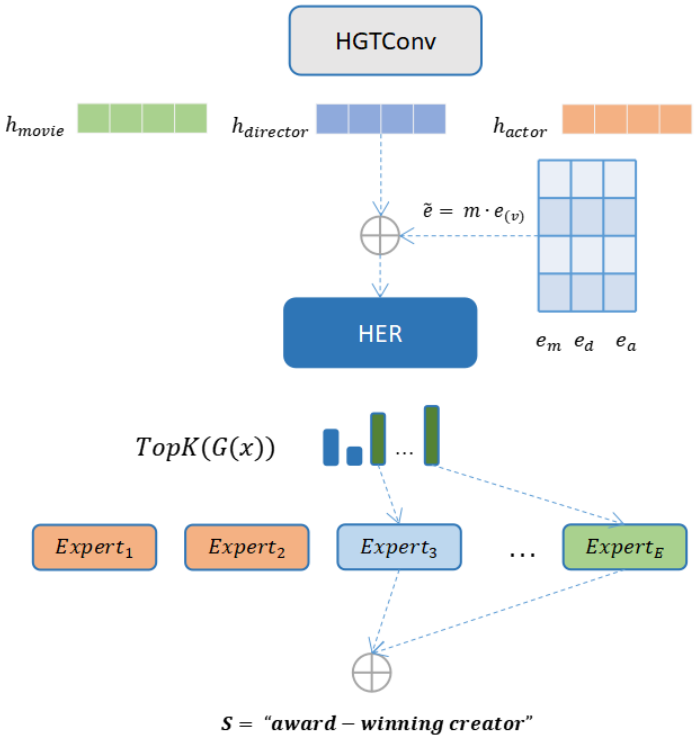

Figure 1: **Mixture-of-Experts in Deep Learning.** The router receives a fused representation $\tilde{\mathbf{x}}_v$, where the type embedding component is stochastically masked during training. A shared set of experts processes inputs from all node types, and the final output is a sparse weighted sum. This shared expert pool enables cross-type semantic transfer–semantically similar entities, regardless of type, can be routed to the same experts.

As illustrated in Figure 1, the SharedMoE with Homogeneous Expert Routing (Definition 3) enables cross-type knowledge transfer by sharing a common set of experts across all node types. This design allows semantically similar entities–regardless of their node type–to be routed to the same experts, fostering the emergence of latent semantic relations through the composition of specialized expert functions. In effect, the model learns to represent heterogeneous concepts (e.g., "award-winning creator") not through type-specific parameters, but via adaptive combinations of profiled experts, thereby enhancing generalization and semantic coherence.

4 Information-Theoretic View of Homogenizing Routing

We formalize two failure modes that arise when node-type information is used naively in MoE-based heterogeneous graph learning.

First, in *SeparatedMoE* (Definition 1), the expert set is partitioned into disjoint subsets $\{E_\tau\}_{\tau \in \mathcal{T}}$, and each node v of type $\tau(v)$ is restricted to experts in $E_{\tau(v)}$. This design eliminates any shared parameter space across types, which—despite rich cross-type representations produced by HGT—prevents *expert-level knowledge transfer*. Even if nodes of different types exhibit identical semantic behavior (e.g., "frequent collaborators" among actors and directors), they are processed by unrelated experts, forcing redundant learning and limiting generalization. Crucially, because routing is fixed by architecture rather than learned from data, SeparatedMoE cannot adapt to semantic similarities that transcend type boundaries.

Second, and more subtly, a naïve instantiation of *SharedMoE*—where the router receives *unmasked* type embeddings—can still exploit a *semantic shortcut*. Let S denote a latent semantic pattern shared across types (e.g., "award-winning creator"), $\mathbf{X} \in \mathbb{R}^{N \times d}$ the matrix of HGT-processed node embeddings encoding structural and feature context, and $\mathbf{T} \in \mathbb{R}^{N \times d}$ the matrix of explicit type embeddings.

Since $I(S;\mathbf{T}) > 0$ (types correlate with semantics), a router that conditions on both $\mathbf{X}$ and $\mathbf{T}$ may achieve low training loss by using $\mathbf{T}$ as a proxy for S, even if it ignores the true semantic signal in $\mathbf{X}$.

Formally, consider a routing function $R(\mathbf{X}, \mathbf{T})$ that has unrestricted access to $\mathbf{T}$. The mutual information between routing decisions and semantics satisfies

$$I\big(R(\mathbf{X}, \mathbf{T}); S\big) \geq I\big(R(\mathbf{T}); S\big), \tag{18}$$

and optimization may converge to a degenerate solution where $R(\mathbf{X}, \mathbf{T}) \approx R(\mathbf{T})$—i.e., expert assignment becomes effectively deterministic in $\mathbf{T}$, nullifying the benefits of a shared expert pool.

To eliminate this shortcut while preserving the capacity for type-aware reasoning when beneficial, we instantiate SharedMoE with *Homogeneous Expert Routing* (HER, Definition 3): during training, each type embedding $\mathbf{e}_{\tau(v)}$ is zeroed out with probability p_{mask}, yielding a stochastically masked variant $\widetilde{\mathbf{T}} = \mathbf{M} \odot \mathbf{T}$, where $\mathbf{M} \in \{0,1\}^{N \times d}$ and $M_{v,:} \sim \text{Bernoulli}(1 - p_{\text{mask}})$. The router thus operates on $(\mathbf{X}, \widetilde{\mathbf{T}})$. By the law of total expectation over the masking variable $\mathbf{M}$, the mutual information between routing decisions and semantics decomposes as:

$$I\big(R(\mathbf{X}, \widetilde{\mathbf{T}}); S\big) = (1 - p_{\text{mask}}) \cdot I\big(R(\mathbf{X}, \mathbf{T}); S\big) + p_{\text{mask}} \cdot I\big(R(\mathbf{X}); S\big). \tag{19}$$

Since mutual information is non-negative, it follows that

$$I\big(R(\mathbf{X}, \widetilde{\mathbf{T}}); S\big) \geq p_{\text{mask}} \cdot I\big(R(\mathbf{X}); S\big). \tag{20}$$

Therefore, if the model achieves high task performance—which requires strong alignment between routing and semantics—it must ensure $I(R(\mathbf{X}); S) > 0$ whenever $p_{\text{mask}} > 0$. In other words, *the router is compelled to extract semantic signals from* $\mathbf{X}$ *itself*, rather than relying on $\mathbf{T}$ as a surrogate.

This mechanism preserves the advantages of a *shared expert space*—enabling cross-type knowledge transfer and semantic role emergence—while actively regularizing against type-determined routing. The result is a routing policy that is *homogenizing* in expectation: it treats nodes as members of a unified semantic space, using type only as a stochastic, non-deterministic cue. This policy is precisely instantiated by Homogeneous Expert Routing (HER), as defined in Definition 3.

5 Experiments

We evaluate our proposed HGT-SharedMoE with Homogeneous Expert Routing (HER) (Definition 3) on three standard heterogeneous graph benchmarks: **IMDB**, **ACM**, and **DBLP** [Hu et al., 2020, Wang et al., 2022]. These datasets are widely used for evaluating relational learning across multiple node and edge types.

The **IMDB** graph contains three node types—*movie*, *director*, and *actor*—and two edge types: *actor–movie* and *director–movie*. The **ACM** graph includes *paper*, *author*, and *subject* nodes connected via *paper–author* and *paper–subject* edges. The **DBLP** graph consists of *paper*, *author*, *venue*, and *term* nodes with corresponding co-occurrence relations. In all cases, we formulate the task as *link prediction*: given a partial graph, predict the existence of masked edges between node pairs of specific type combinations (e.g., actor–movie in IMDB). We use ROC-AUC as the primary evaluation metric.

Our experiments compare three model variants:

1. **HGT (baseline)**: Standard HGT without MoE.
2. **SeparatedMoE**: Type-separated MoE layers, where each node type has its own disjoint set of experts.
3. **SharedMoE (ours)**: MoE layer with HER, trained with stochastic type masking at rate p_{mask}.

In our experiments, each node type has access to $K = 16$ experts. Specifically, **SharedMoE** employs a single shared pool of $K = 16$ experts across all node types, whereas **SeparatedMoE** assigns a dedicated set of $K = 16$ experts *per node type*, resulting in a total of $|\mathcal{T}| \times E$ experts (e.g., 48 for IMDB with $|\mathcal{T}| = 3$). Despite having significantly fewer parameters, **SharedMoE** consistently outperforms **SeparatedMoE**—demonstrating that cross-type parameter sharing leads to more efficient and generalizable representations.

All models share an identical HGT backbone with hidden dimension $d = 64$, HGT layers, 4 attention heads per layer, and a balance loss weight of $\alpha = 0.01$. For SharedMoE, we sweep $p_{\text{mask}} \in \{0.0, 0.1, 0.3, 0.5, 0.8, 0.9, 1.0\}$. The results are reported as mean ± standard error of the mean (SEM) over 10 random seeds.

5.1 Link Prediction Performance

Results across all datasets are summarized in Table 1. On IMDB, SharedMoE with $p_{\text{mask}} = 0.3$ achieves 92.63% ROC-AUC, surpassing both HGT (+5.06%) and SeparatedMoE (+2.29%). On ACM, SharedMoE reaches 84.97% at $p_{\text{mask}} = 0.1$, outperforming SeparatedMoE (79.22%) by 5.75 percentage points and HGT (no MoE) (75.62%) by 9.35 points. On DBLP, SharedMoE maintains consistently high performance, peaking at 91.99% ($p_{\text{mask}} = 0.3$), which exceeds SeparatedMoE (89.27%) by 2.72 points and HGT (no MoE) (87.68%) by 4.31 points. The non-monotonic dependence on p_{mask}—with optimal performance at moderate masking and severe degradation at $p_{\text{mask}} = 1.0$—demonstrates that partial stochastic masking effectively regularizes expert routing without erasing essential type information.

Table 1: Link prediction performance (ROC-AUC ± standard error) across IMDB, ACM, and DBLP benchmarks. SharedMoE achieves peak performance at $p_{\text{mask}} = 0.3$ on IMDB (92.63%) and DBLP (91.99%), while on ACM the best result occurs at $p_{\text{mask}} = 0.1$ (84.97%). SharedMoE consistently outperforms both the HGT (no MoE) baseline and SeparatedMoE across all datasets. Performance degrades sharply at $p_{\text{mask}} = 1.0$, confirming that moderate masking provides optimal regularization.

Model	**IMDB**	**ACM**	**DBLP**
HGT (no MoE)	87.57 ± 0.52	75.62 ± 2.81	87.68 ± 1.05
SeparatedMoE	90.34 ± 0.58	79.22 ± 3.43	89.27 ± 1.00
SharedMoE ($p_{\text{mask}} = 0.0$)	89.88 ± 0.67	79.42 ± 2.40	91.59 ± 0.55
SharedMoE ($p_{\text{mask}} = 0.1$)	92.10 ± 0.24	**84.97 ± 3.10**	91.64 ± 0.30
SharedMoE ($p_{\text{mask}} = 0.3$)	**92.63 ± 0.48**	80.57 ± 3.13	**91.99 ± 0.31**
SharedMoE ($p_{\text{mask}} = 0.5$)	91.87 ± 0.54	80.28 ± 1.45	91.76 ± 0.55
SharedMoE ($p_{\text{mask}} = 0.8$)	90.13 ± 0.73	71.65 ± 2.68	91.55 ± 0.32
SharedMoE ($p_{\text{mask}} = 0.9$)	90.51 ± 0.90	72.42 ± 2.98	91.79 ± 0.32
SharedMoE ($p_{\text{mask}} = 1.0$)	70.12 ± 2.21	73.54 ± 2.27	78.81 ± 0.87

5.2 Expert Activation Analysis

To validate our hypothesis that a shared expert pool enables semantic-driven specialization, we analyze expert activation patterns across node types on the **IMDB** heterogeneous graph benchmark. All analyses correspond to the best-performing SharedMoE model ($p_{\text{mask}} = 0.3$) trained for link prediction on IMDB.

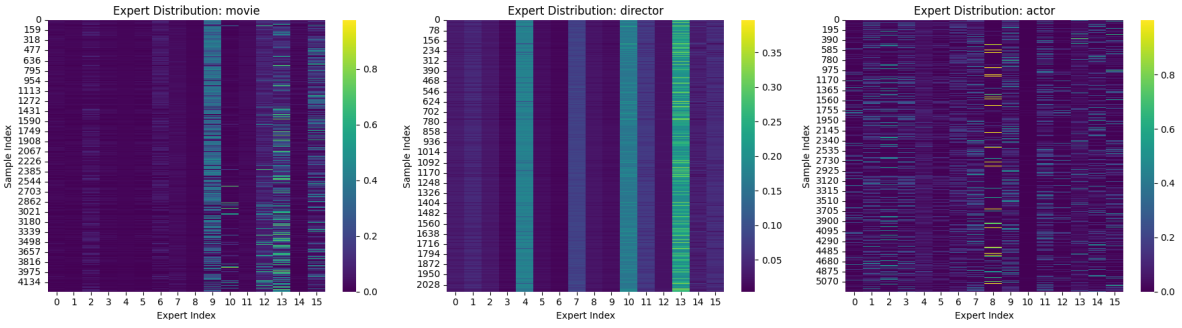

Figure 2: Expert activation heatmap for SharedMoE ($p_{\text{mask}} = 0.3$) on IMDB. Rows correspond to node samples (grouped by type: actor, director, movie); columns represent expert indices (0–15). Color intensity reflects the routing weight assigned to each expert. Experts 9–10 and 12–15 consistently dominate across all node types, indicating that routing is driven by shared semantic cues rather than type identity.

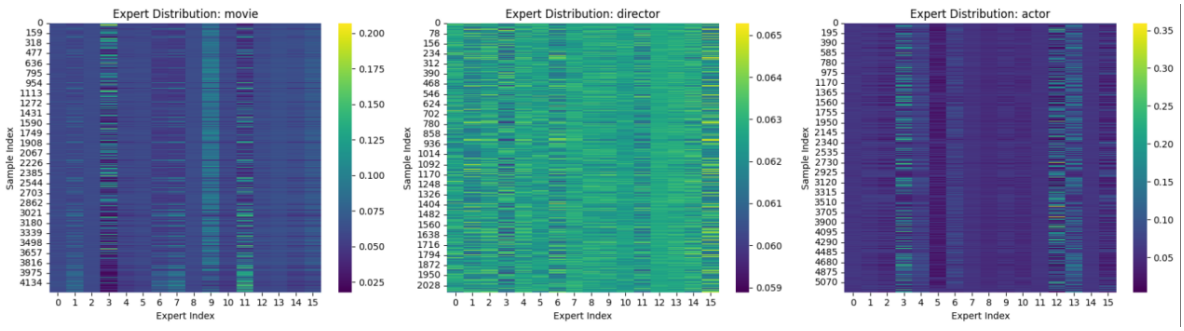

Figure 3: Expert activation heatmap for SeparatedMoE on IMDB. Each node type uses its own disjoint expert set. Activations are diffuse and uniformly distributed within each type, with no dominant expert emerging—suggesting fragmented specialization and lack of cross-type alignment.

Figure 2 shows that in SharedMoE, all three node types—actor, director, and movie—consistently activate the same subset of experts (notably indices 12–15), while assigning negligible weight to others. This strong cross-type alignment demonstrates that the router conditions its decisions on latent semantic signals present in the HGT-processed embeddings, not on node-type labels.

In contrast, Figure 3 reveals that SeparatedMoE—despite receiving the same rich HGT representations—exhibits diffuse and uniform activation within each type-specific expert pool. No expert dominates for any type, indicating that the model fails to develop coherent functional roles. This is not due to a lack of semantic information (which is already encoded by HGT), but because the architectural separation prevents parameter sharing: even if actors and directors exhibit similar collaborative patterns, they must be modeled by distinct, uncoordinated experts. The result is redundant learning and weak specialization.

These findings, specific to the IMDB benchmark, confirm that the advantage of SharedMoE lies not in superior access to graph semantics, but in its *shared parameter space*, which allows a single expert to capture a semantic role (e.g., "prolific collaborator") across all node types—leading to more efficient, coherent, and generalizable representations.

5.3 Semantic Specialization of Experts: Correlation with External Labels

Although our model is trained solely for unsupervised link prediction on IMDB and never exposed to high-level semantic labels, we observe that experts in SharedMoE implicitly specialize along interpretable, external semantic dimensions. This emergent structure—analyzed exclusively on the IMDB dataset—provides strong evidence that routing is governed by latent semantics, not syntactic type identity.

To quantify this, we analyze the distribution of movie genres across experts using the IMDB genre annotations. For each expert i, we compute the normalized frequency of assigned movies belonging to each genre (Action, Comedy, Drama). Results in Table 2 reveal pronounced genre biases: Experts 0 and 9 are strongly associated with Comedy (57% and 60%), Experts 12 and 13 with Drama (60% and 57%), and Expert 14 with Action (43%). This indicates that experts learn to encode genre-related functional roles—such as "comedy-oriented project"—without any explicit supervision.

It is important to clarify the causal direction of this phenomenon: HER does not *create* semantic roles de novo. Experts serve as interpretable *functional units* that *capture and amplify* semantic signals already present in HGT embeddings, by allowing nodes with similar semantic context—regardless of type—to be processed by the same subnetworks.

Crucially, this semantic coherence extends beyond movies to other node types. We find that *directors are predominantly routed to Expert 13* as their top-1 expert (Figure 4) . Expert 13 also processes a substantial fraction of drama-themed movies (57%) and their associated actors, making it one of the most drama-biased expert in the pool.

One might hypothesize that this routing arises from *topological similarity* among directors—e.g., if directors formed densely connected subgraphs or shared common structural motifs, a router could learn to assign them collectively based on syntactic graph features alone. However, in the IMDB graph, directors are *not directly connected* to one another, and each movie is typically associated with a *single* director. Consequently, directors appear as structurally isolated leaf nodes whose local

neighborhoods consist almost exclusively of disjoint sets of movies and actors. Under such sparse and heterogeneous connectivity, a purely structural router would be expected to distribute directors across multiple experts, reflecting the diversity of their contexts.

The fact that directors instead *converge* on a single expert—Expert 13—that is also selected by drama movies and drama-associated actors strongly suggests that routing is driven not by graph topology, but by a *latent semantic signal*: namely, participation in drama-related productions. This interpretation is further supported by the structural properties of the IMDB graph: directors are not directly connected to one another, and each typically appears in only a single movie node's neighborhood, resulting in sparse, heterogeneous, and largely non-overlapping local structures. Under such conditions, a router relying solely on topological or syntactic cues would be expected to distribute directors across many experts, reflecting the diversity of their collaboration contexts. The observed convergence—despite this structural fragmentation—rules out graph topology as the primary driver and implies that HER's router grounds its decisions in content-rich signals from HGT-processed embeddings. Consequently, this cross-type alignment around a shared genre concept cannot be explained by local structural cues or superficial type–genre correlations alone, and provides compelling evidence that HER induces *semantic role specialization* that transcends both type identity and syntactic graph position.

To visualize how these roles manifest across types, we project final node embeddings using t-SNE, coloring by top-1 assigned expert and shaping by node type (Figure 4). Nodes of different types—movies, actors, and directors—frequently co-locate within the same expert cluster. For example, Expert 12 (purple) groups drama films together with their associated cast and crew, while Expert 9 (blue) clusters comedy-related entities. This cross-type co-clustering confirms that expert assignment reflects shared semantic context, not categorical type. Because all types share the same expert pool, the model can assign a drama actor, a drama film, and a drama director to the same functional unit—enabling explicit alignment of semantic roles. In contrast, SeparatedMoE cannot achieve such alignment by construction, even if its internal representations encode the same signals.

Thus, on the IMDB benchmark, SharedMoE's superiority stems from its ability to *organize expert specializa around semantic roles* rather than syntactic categories—a capability made possible by parameter sharing and homogenizing routing.

Table 2: Genre distribution per expert among assigned movies on IMDB. Experts 0 and 9; 12 and 13; and 14 show pronounced specialization toward Comedy, Drama, and Action, respectively.

Expert	**Action**	**Comedy**	**Drama**
0	30%	**57%**	13%
1	20%	30%	50%
2	20%	33%	47%
3	13%	40%	47%
4	33%	27%	40%
5	37%	23%	40%
6	30%	43%	27%
7	23%	47%	30%
8	10%	47%	43%
9	13%	**60%**	27%
10	17%	40%	43%
11	27%	43%	30%
12	13%	27%	**60%**
13	23%	20%	**57%**
14	**43%**	22%	33%
15	27%	47%	26%

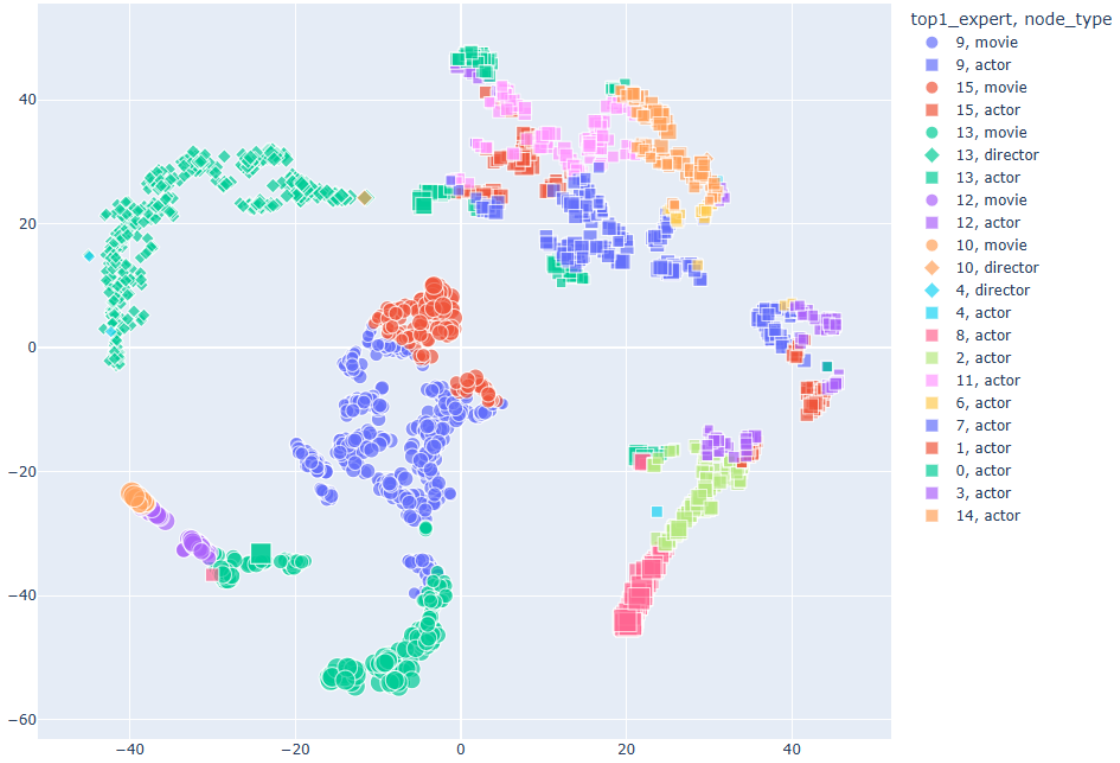

Figure 4: **t-SNE visualization of node embeddings on IMDB, colored by top-1 expert and shaped by node type.** Clusters correspond to expert specializations: e.g., Expert 9 (blue) is enriched with Comedy films, Expert 12 (purple) with Drama. The co-location of movies, actors, and directors within the same expert cluster confirms that routing is driven by semantics, not type.

5.4 Expert Activation Dynamics under Type Masking

To empirically validate our core hypothesis—that HER encourages experts to specialize on semantic patterns shared across node types, rather than on type-specific labels—we conduct a quantitative analysis of expert activation distributions **on the IMDB dataset**. We evaluate SharedMoE models trained with stochastic type masking across the full range of rates $p_{\text{mask}} \in \{0.0, 0.1, 0.3, 0.5, 0.8, 0.9, 1.0\}$. For each model, we extract the softmax routing probabilities over the $K = 16$ experts for all nodes and compute the following metrics:

1. **Cross-type cosine similarity**: average cosine similarity between mean expert activation vectors of different node types. Higher values indicate more consistent expert usage across types.

2. **Normalized entropy of expert assignment per node type**: $H(T)$. Lower entropy indicates sparse, confident routing.

3. **Top-2 coverage**: fraction of nodes for which the top-2 experts receive at least 80% of the total routing weight. Measures per-node routing sparsity and expert specialization.

The complete results are presented in Table 3. The data reveals a non-monotonic relationship between masking rate, cross-type alignment, and routing sparsity. Peak performance (AUC = 92.63%) is achieved at $p_{\text{mask}} = 0.3$, which represents an optimal trade-off: cross-type similarity (0.606) reflects meaningful semantic alignment, while routing remains sufficiently sparse—e.g., top-2 coverage for actors reaches 0.208, and movie routing retains meaningful concentration (0.125).

In contrast, the model with no masking ($p_{\text{mask}} = 0.0$) exhibits a high cross-type similarity of 0.705, but this value is misleading: it arises from deterministic, type-driven routing (e.g., all actors routed to expert A, all directors to B), which creates artificial alignment without semantic grounding. Such behavior fails to generalize to unseen type configurations.

At the other extreme, full masking ($p_{\text{mask}} = 1.0$) yields the highest numerical similarity (0.824), yet this reflects a degenerate regime: with type information completely removed, the router collapses to near-uniform expert assignment (entropy > 0.92 for all types, top-2 coverage $= 0.0$). The resulting lack of specialization explains the catastrophic performance drop (AUC = 70.12%).

Notably, directors consistently exhibit high entropy and zero top-2 coverage across all masking rates, suggesting their representations—while enriched by HGT—lack discriminative semantic structure in this task setting, possibly due to sparser connectivity or weaker genre correlation.

These findings, derived from IMDB, confirm a central insight: *optimal performance arises not from maximal type invariance nor from rigid type dependence, but from a balance between semantic alignment and routing fidelity.* Stochastic masking at moderate rates ($p_{\text{mask}} = 0.3$) regularizes the router away from type shortcuts while preserving enough signal to enable sparse, confident expert selection—precisely the regime where shared parameterization yields its greatest benefit.

Table 3: Full expert activation metrics for SharedMoE under varying type masking rates on IMDB. The optimal performance at $p_{\text{mask}} = 0.3$ coincides with a balance between cross-type alignment and routing sparsity.

p_{mask}	Cross-type Sim.	H(M)	H(D)	H(A)	Top-2 Cov. (M)	Top-2 Cov. (D)	Top-2 Cov. (A)
0.0	0.705	0.690	0.916	0.704	0.072	0.000	0.243
0.1	0.583	0.484	0.891	0.550	0.362	0.001	0.345
0.3	**0.606**	**0.652**	**0.919**	**0.629**	**0.125**	**0.000**	**0.208**
0.5	0.484	0.779	0.891	0.586	0.018	0.000	0.184
0.8	0.573	0.777	0.973	0.718	0.095	0.000	0.089
0.9	0.369	0.822	0.908	0.592	0.001	0.000	0.003
1.0	0.824	0.956	0.971	0.929	0.000	0.000	0.000

6 Conclusion

We have presented *Homogeneous Expert Routing* (HER)—a novel MoE-based architecture for heterogeneous graph learning that decouples expert specialization from syntactic node-type identity. While MoE has achieved remarkable success in homogeneous settings [Zhang et al., 2023], its adaptation to heterogeneous graphs requires careful handling of type metadata to avoid semantic shortcuts or redundant modeling.

HER addresses this by combining a *shared expert pool* with a *regularized router* that conditions on HGT-processed node representations and stochastically masked type embeddings. This design preserves the capacity to exploit type information when beneficial, while forcing the router to rely on latent semantic content during training—thereby encouraging experts to specialize on type-agnostic functional roles.

Evaluated on three standard benchmarks (IMDB, ACM, DBLP), HER consistently outperforms both standard HGT and a type-separated MoE baseline in link prediction, despite using significantly fewer parameters. Our qualitative analysis on IMDB reveals that HER induces genuine cross-type semantic alignment: movies, actors, and directors participating in drama or comedy productions are co-routed to the same experts, and expert assignments strongly correlate with external genre labels. This behavior—structurally impossible under SeparatedMoE—demonstrates that HER enables experts to capture relational semantics that transcend categorical type boundaries.

These results establish that in heterogeneous graph learning, expert routing benefits from a shared parameter space and controlled, regularized use of type information. HER provides a simple yet effective instantiation of this principle, opening avenues for dynamic graphs, cold-start generalization, and interpretable role discovery in structured relational data.

References

Y. Dong, N. V. Chawla, and A. Swami. Metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge*

Discovery and Data Mining, pages 135–144, 2017.

W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin. Graph neural networks for social recommendation. In *The World Wide Web Conference (WWW)*, pages 417–426, 2019.

W. Fedus, B. Zoph, and N. Shazeer. Switch transformers: Scaling to trillion parameter models. *Journal of Machine Learning Research*, 23(1):1–45, 2022a. URL `https://www.jmlr.org/papers/v23/21-0537.html`.

W. Fedus, B. Zoph, N. Shazeer, et al. Glam: Efficient scaling of language models with mixture-of-experts. *arXiv preprint arXiv:2112.06905*, 2022b.

Z. Hu, Y. Dong, K. Wang, K.-W. Chang, and Y. Sun. Heterogeneous graph transformer. In *Proceedings of The Web Conference 2020 (WWW)*, pages 2704–2710, 2020.

K. Huang et al. Unignn: A unified framework for graph neural networks on heterogeneous graphs. In *AAAI Conference on Artificial Intelligence*, 2022.

R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

Z. Liu et al. Discovering semantic roles in social networks via graph neural networks. In *Proceedings of the ACM Web Conference (WWW)*, 2022.

X. Lv et al. Simple-hgn: Simplifying heterogeneous graph networks. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2562–2568, 2021.

J. Puigcerver et al. Scalable vision transformers with hierarchical mixture of experts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14158–14167, 2021.

M. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, pages 593–607. Springer, 2018.

N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017.

K. Wang, Z. Hu, et al. Hgb: A comprehensive benchmark for heterogeneous graph learning. In *Proceedings of the ACM Web Conference (WWW)*, 2022.

Q. Wang, Z. Mao, B. Wang, and L. Guo. Knowledge graph embedding: A survey of approaches and applications. In *IEEE Transactions on Knowledge and Data Engineering*, volume 29, pages 2785–2802, 2019.

Y. Wang et al. Inductive representation learning in temporal networks via causal anonymous walks. In *International Conference on Learning Representations (ICLR)*, 2021.

C. Zhang, D. Song, C. Huang, A. Swami, et al. Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 793–803, 2019.

Y. Zhang, X. Wang, et al. Graph mixture of experts. In *International Conference on Machine Learning (ICML)*, 2023.