CausGT-HS: An Energy-Based Causal MoE-GNN for Causal Reasoning

Siddharth Palod
IMT2022002
IIIT-Bangalore

Shashank Tippanavar
IMT2022014
IIIT-Bangalore

Kushal Jenamani
IMT2022057
IIIT-Bangalore

Shanmukh Praneeth
IMT2022542
IIIT-Bangalore

***Abstract*—Traditional knowledge graphs and standard graph neural networks are purely correlational, and discovering latent causal mechanisms from static, observational text data is challenging. Traditional causal discovery methods do not solve this problem. This paper introduces a Causal GT-HS, an energy-based, dual-stream, hierarchical-sparse graph transformer that converts noisy, weighted, multi-relational, correlational graphs extracted from documents into clean, directed causal meta-path graphs. The architecture features end-to-end weighted meta-path learning using a Graph Transformer network, scalable hierarchical coarsening to address $O(N^2)$ bottlenecks, and a dual-stream attention mechanism with dynamic gated fusion to differentiate correlational from causal signals, enabling self-supervised inference of explanatory causal structures over large-scale document graphs.**

***Index Terms*—Knowledge Graphs, Graph Transformer, Large Language Models, Retrieval-Augmented Generation, Graph Neural Networks**

I. Introduction

The central goal of advanced information retrieval is to move beyond simple, correlational (which has to do only with geometry) extraction to discover deep, explanatory, and **causal** mechanisms hidden within unstructured text (e.g., PDFs). Current systems, like GraphRAG are all ”correlational engines”, meaning they excel at identifying and summarizing explicitly stated relationships (i.e., just the retrieved facts), effectively answering **”what”** and not **”why”**. Our objective is to build a system that answers **”Why is this related?”** by discovering the underlying causal meta-paths.

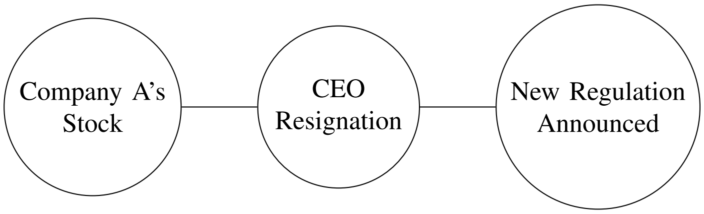
- latent = hidden
- meta-paths = path of classes and not instances.
- causal = seeing the cause and not just structural similarity.

In simple terms, the problem is the difference between **”what”** and **”why”**.

Today we are excellent at building KGs that tell us what things are related. But we are terrible at building KGs that tell us why they are related or which ”what” caused the other. This is the **”correlation vs causation”** problem.

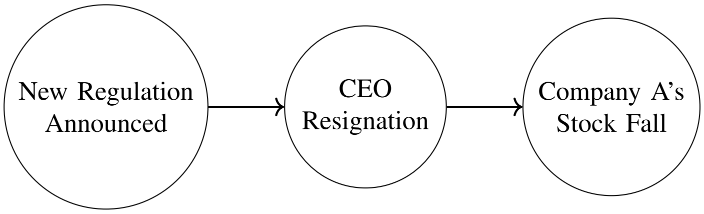
Let us see this with an example: `A financial news PDF` Imagine an article in the newspaper about some company.

- **Current KGs (That ”What”/Correlation):** Your system can easily extract a graph like this:



This graph is a “dumb” map. It is flat and undirected. It only tells you that these topics appeared together, but it offers no explanation.
Did the stock fall because the CEO resigned? Or did the new regulation cause the CEO to resign, which then caused the stock to fall? A standard Knowledge Graph (KG) cannot tell the difference.

- **Our Goal (The ”Why”/Causation):** We want to automatically discover the real story - the **causal meta-path** A possible causal chain can be represented as:



Unlike an undirected knowledge graph, this causal structure indicates how one event may influence another.

The Core Challenge

How can a computer discover this ”why” graph ($A \to B$) when all it has to read is a single, static PDF?? This is called a **”static observational data trap”**. All the computer sees is that A and B are ”observed” together, not which one ”acted” first.

Static observational data trap

Refers to the methodological pitfalls and limitations that arise when researchers use data collected at a single point in time to make conclusions that require information about change, cause, or dynamic processes.

This presents a formidable challenge, which existing methods are unequipped to solve:

1) **The Observational Data Trap:** The gold standard methods for causal discovery (e.g., DAG-GNN, NOTEARS, PC-Algorithm) are designed for tabular, interventional, or time-series data. They require seeing how variables change in response to each other. Document KGs are **static and purely observational**. We only have one ”snapshot” of the graph derived from static text. Causality must be inferred from static linguistic cues and world knowledge, not observed changes.
2) **The ”Causality-Blind” GNNs:** Standard GNNs (GCN, GAT) are fundamentally ”correlation” based. Their message-passing mechanisms (even with attention) are typically symmetric (same in both directions of edges). They cannot, by design, differentiate between $A \to B$ (causation) and $A - B$ (correlation).
3) **The Noise of Reality:** Real-world knowledge is not binary. The initial graph *G(V, E, R)* extracted from a PDF is noisy. A superior model must operate on a **weighted multi-relational graph G(V, E, R, W)**, where W represents initial ”correlational” confidences or like proximity.
4) **The $O(N^2)$ Scalability Bottleneck:** Real-world documents (e.g., a 100-page technical manual or legal filing) can contain $N > 100,000$ unique entities. Any algorithm that requires building or computing $N \times N$ matrices (e.g., a full attention mechanism or a dense adjacency matrix) is **computationally infeasible**. A good solution must scale linearly or near-linearly ($O(N) \quad \text{or} \quad O(N \log N)$).

We propose a novel architecture, **CausGT-W**, that solves all the above challenges by creating a new end-to-end Graph transformer that learns causality by being ”taught” by an LLM's **counterfactual reasoning** to generate a **Causal Prior dataset**. It is a self-supervised framework for discovering directed, causal meta-paths from static, observational text.

We do this by not asking the LLM ”is this causal?” but by forcing it to perform counterfactual reasoning (”What if...?”) on text snippets. For example, ”If Algorithm A were NOT used, what would be the impact on Accuracy?” The answers to these ”what if” questions, which the LLM can infer from its vast world knowledge, become our only supervisory signal for causality.

II. Related Works

A. Causality Extraction from Text using Neural Language Models

Recent neural language models such as RoBERTa [12] and T5 [13] have advanced causal signal extraction from text by enabling richer contextualized representations and by supporting tasks such as implicit relation linking [11]. These models capture lexical, syntactic, and discourse-level indicators of causation and can recover hidden or underspecified cause–effect pairs. Further developments in label-aware contextualization [6] demonstrate that pretrained LMs can learn to differentiate relation types through verbalization-driven finetuning.

However, the mainstream LM-based causal extraction literature remains *correlational*, not *causal*. As noted by recent critiques [45], LLMs frequently hallucinate or invert causal directionality, fail to preserve acyclicity, and cannot guarantee structural validity. Attempts to enforce causal priors by using LLMs as weak supervisors or constraints [44] mitigate some errors but do not equip these models with explicit causal reasoning mechanisms. They continue to lack guarantees regarding graph-level properties such as consistency, minimality, or directed path validity, which are essential for causal discovery from text-derived graphs.

B. GNN- and Path-Based Causal Discovery

Graph neural networks (GCN [31], GraphSAGE [36], and relational GNNs such as R-GCN [8]) have been leveraged for causal link prediction by encoding asymmetric dependencies or directional relational constraints. Recent surveys [49] emphasize extensions of GNNs with attention mechanisms, directional message passing, and disentangled representations aimed at capturing cause–effect patterns.

Probabilistic causal GNN models [46] represent a newer development, introducing uncertainty over edge directions and enabling sampling-based inference of causal structures. Meanwhile, path-based models such as PathNN [35] and the more expressive PathWL [48] show that multi-hop dependency modeling is essential for reconstructing causal chains that span multiple textual events or entity mentions. These works highlight that many causal mechanisms emerge only through multi-hop event propagation, not direct adjacency.

Despite progress, GNNs remain limited by their largely symmetric message-passing nature and by their difficulty in encoding strongly direction-sensitive dependencies. Path-based systems, although more expressive, rely heavily on accurate graph construction from text; errors in early extraction propagate through the entire causal chain. Classical path enumeration techniques such as Yen's algorithm [53] exacerbate this by generating large numbers of spurious candidate paths that require principled causal filtering — something most current neural methods do not provide.

C. Causal Discovery via Knowledge Graph and Temporal Graph Reasoning

Knowledge graph embedding models such as TransE [2], RotatE [3], and QuatE [4] encode relational structure geometrically and can infer missing directional edges. Variants that use hyperplanes, rotations, or manifold transformations [19] improve the modeling of asymmetric relations that resemble causal directionality. However, these methods optimize *relational plausibility*, not *causal validity*, and therefore may treat correlation, entailment, or topical similarity as causal influence.

Temporal reasoning models such as RENet [9] and L2T-KG [10] learn evolving event dependencies and time-sensitive relations, which are useful for causal inference when explicit timestamps exist. Knowledge-graph-based causal path extraction frameworks [43] attempt to isolate subgraphs corresponding to causal reasoning chains by identifying informative or directionally biased subpaths.

Nonetheless, these methods face structural limitations: static KG embeddings lack acyclicity enforcement, temporal models require high-quality timestamps unavailable in raw text, and causal-path extraction approaches often produce dense, noisy subgraphs without principled criteria to distinguish causal mechanisms from correlational or evidential associations.

D. Energy-Based, Score-Based, and Optimization Approaches for Causal Discovery

Energy-based models (EBMs) [29], [32] and recent tabular causal EBMs [47] formulate causal discovery as learning an energy landscape where low-energy configurations correspond to plausible causal graphs. Score-based generative models [27] and Langevin-based samplers [26] provide mechanisms for traversing this landscape, enabling sampling of candidate causal graphs under model uncertainty. Such approaches are highly expressive and can support counterfactual querying within the learned energy geometry.

Complementary to these, continuous graph-optimization approaches like NO-TEARS [33] enforce acyclicity constraints differentiably, enabling gradient-based learning of DAG structures directly. These frameworks have influenced many modern causal discovery pipelines that attempt to impose structure on embeddings or graph encoders derived from text.

However, EBM-based and score-based models struggle with high-dimensional, noisy embeddings typical of text-derived KGs, where energy gradients become unstable. Langevin-based methods assume well-calibrated distributions over variables, which text-derived nodes rarely satisfy. Continuous DAG methods further assume clean variable definitions; ambiguous or multi-granular nodes extracted from text can cause structurally invalid or uninterpretable causal outputs. Hybrid approaches [44] that incorporate external supervision partially alleviate these issues but do not resolve them fundamentally.

III. Methodology

Our model architecture is divided into two stages:
1. Mediator-Controlled Causal Prior (C_{prior}) Generation
2. CausGT-HS: A self-supervised, probabilistic energy-based causal graph-token transformer
The detailed architecture is illustrated in Fig. 1.

A. Model Information

1) Model inputs::

- **Node Features:**

$$H^{(0)} = X \in \mathbb{R}^{X \times d_{in}}$$

, where N = number of nodes (entities) and d_{in} is the initial embedding dimension.

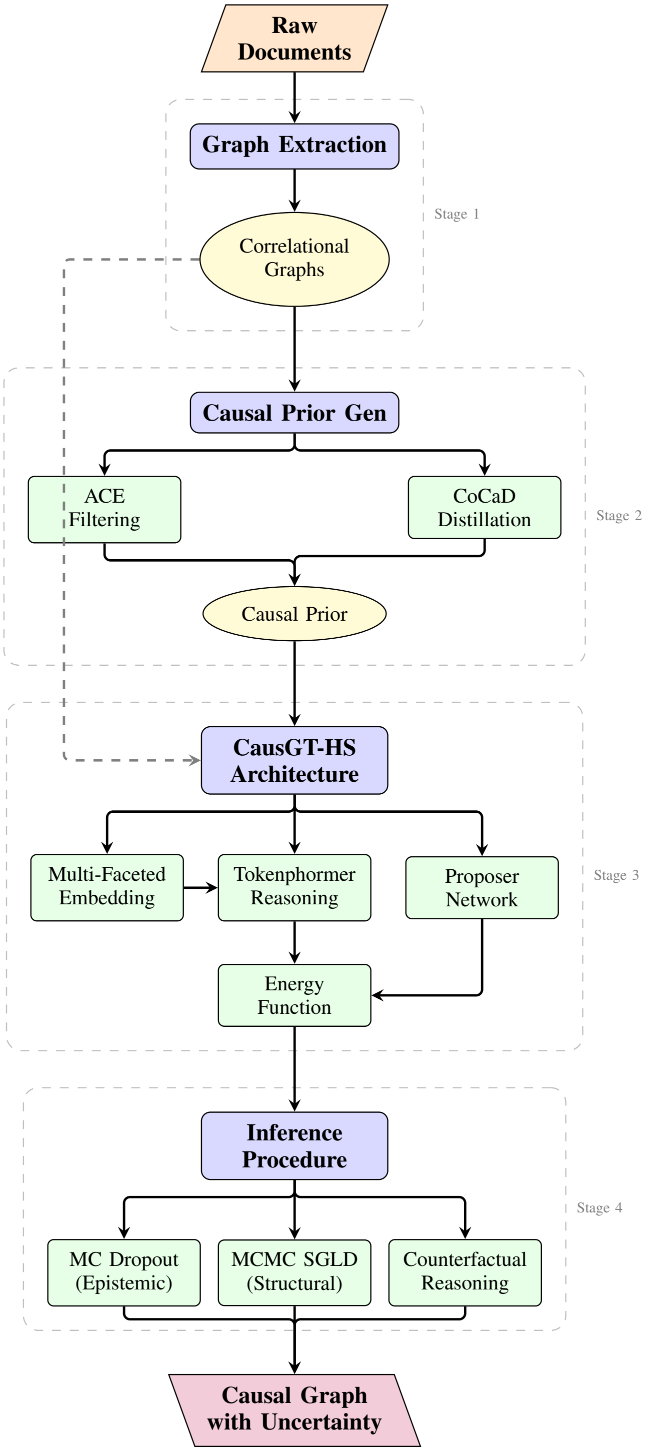


Fig. 1. Proposed Pipeline of CausGT-HS.

- **Weighted Relational Graphs:** A set of K weighted adjacency matrices

$$A_W = \{W_1, W_2, \ldots, W_K\} \quad \text{where} \quad W_r \in [0,1]^{N\times N}$$

Each one of these matrices above matches to one type of single, correlational relationship. Each of the W_r above represents relationships between any of the N nodes and any of the other N nodes for a specific relation r. $|E_r|$ is the number of non-zero entries in W_r. Let $|E|$ = $\sum |E_r|$.

- **The Causal Prior** C_{prior}: Our GNN is supposed to learn to immitate this. This stores a causal score for every possible directed pair from node i to node j.

2) Model outputs:

- **The primary output:** The causal meta-path graph $W_{\text{CausalMeta}} \in [0,1]^{N\times N}$ that represents the underlying causal mechanisms. They represent the final learned causal strength for any directed path from node i to node j.
- **The secondary output:** Our GNN's other job was to update node features. Hence an another output is $H^{(final)}$, which are our causal aware node embeddings. This is the final list of node embeddings after they have passed through all the causal transformer layers. (So now these embeddings are causal-aware).

B. Correlational Graph Extraction

To construct A_W from raw document text D, our LLM-guided preprocessing pipeline first segments D into paragraphs P_p and sentences s_m, performs LLM-NER to obtain the node set $\mathcal{N}$, and builds an inverted index T_{map} to unify entity mentions across the document. For each paragraph P_p, we collect its local entity set N_p and form candidate co-occurrence pairs

$$E_{probe} = \bigcup_p \{(i,j) \mid i,j \in N_p\},$$

capturing correlational links without the $O(|\mathcal{N}|^2)$ explosion.

Paragraph batches G_p are submitted to the LLM to classify surface relations and estimate confidence scores. These outputs populate sparse COO-format matrices

$$W_k = (\text{rows}_k,\ \text{cols}_k,\ \text{data}_k),$$

yielding a multi-relational weighted graph. We further refine edge weights using mention frequency, contextual fit, and LLM confidence to suppress weak or noisy associations. After filtering via cross-paragraph consistency and entropy-based thresholds, relation-specific matrices are combined into

$$A_W = \sum_k \alpha_k W_k,$$

a normalized correlational graph encoding the document's surface-level relational structure.

This A_W forms the input to the ACE module, which identifies structurally informative edges and prunes spurious correlations before causal refinement in CoCaD and the downstream energy-based causal learner.

C. Mediator-Controlled Causal Prior (C_{prior}) Generation

Our goal here is to create the sparse "answer key" C_{prior} by efficiently and intelligently querying the teacher LLM.

Our C_{prior} is a high-quality "training dataset" that tells us the true causal links in the document. Since we have no human labels, we must generate the dataset ourselves. This is a self-supervised process. The core idea is to use a large, powerful "Teacher" LLM to perform complex causal reasoning. The output of this phase is the C_{prior} (Causal Prior).

C_{prior}

The C_{prior} is a sparse training dataset of direct causal effect scores.

- **Form:** It is a list of tuples (i, j, score), where i is a *subjectnodeid*, j is an *objectnodeid*, and score is a float $\in [0.0, 1.0]$.
- **Purpose:** It serves as the supervisory signal (or "ground truth") for the L_{causal} (Causal Loss) function.
- **Meaning:** A score of 1.0 means the "Teacher" LLM has determined there is a direct causal link $i \to j$. A score of 0.0 means there is no direct link.
- **Process:** Our CausGT-HS GNN (the "Student" model) is trained to replicate these scores. This process is known as **Knowledge Distillation**, where we distill the slow, complex, symbolic reasoning of the "Teacher" LLM into the fast, numerical, graph-based architecture of the "Student" GNN.

This stage is further divided into two steps:
(i) Active Candidate-Set Expansion (ACE)
(ii) CoCaD (Counterfactual Causal-DP)

1) Active Candidate-Set Expansion (ACE): - Direct evaluation of N^2 possible node pairs by an LLM for causal reasoning is computationally impossible (e.g., 2.5 billion queries for N=50k). ACE addresses this via a multi-stage filtering cascade that prunes the search space to a small, high-recall list of candidate pairs E_{prior}.

a) Structural Filter (Graph Autoencoder (GAE)): The initial correlational matrices A_W capture only local 1-hop relationships within paragraphs, missing multi-hop structural connections. A Graph Autoencoder (GAE) discovers these latent paths unsupervised. First, we construct the binary co-occurrence scaffold $A_{\text{co-occur}} = \bigvee_k W_k \in \{0,1\}^{N\times N}$. The 2-layer GCN encoder processes node features X and $\tilde{A} = A_{\text{co-occur}} + I$:

$$\begin{aligned} H^1 &= \text{ReLU}\left(D^{-1/2}\tilde{A}D^{-1/2}XW_0\right), \\ Z &= D^{-1/2}\tilde{A}D^{-1/2}H^1W_1 \end{aligned} \tag{1}$$

where $W_0 \in \mathbb{R}^{d_{in}\times d_h}$, $W_1 \in \mathbb{R}^{d_h\times d_z}$ ($d_z = 64$), using sparse matrix multiplication (SpMM, $O(Ed_h)$). The decoder predicts link probabilities via inner products $S_{\text{GAE}} = ZZ^T$ (logical). GAE trains on sampled binary cross-entropy:

$$L = \sum_{(i,j)\in\mathcal{P}} \log \sigma(Z_i^T Z_j) + \sum_{(i,k)\in\mathcal{N}} \log(1 - \sigma(Z_i^T Z_k)) \tag{2}$$

($\mathcal{P}$: observed edges, $\mathcal{N}$: sampled negatives). Candidates are extracted via FAISS index on Z: for each i, retrieve top-k neighbors yielding $C_1 = \bigcup_i \{(i,j) \mid j \in \text{Neighbors}(Z_i, k)\}$.

b) Semantic Filter (RAG-HyDE-RAV): GAE-discovered pairs may be structurally plausible but semantically nonsensical (e.g., unrelated stocks co-mentioned in news). The semantic filter uses RAG-HyDE with Retrieval-Augmented Verification (RAV).

For each $(i,j) \in C_1$, an LLM generates k_{hyp} hypothetical causal sentences H=h_1 describing the plausible $i \rightarrow j$ links. Document sentences $V_D = f_{\text{embed}}(S)$ form a FAISS index. For each hypothesis h_1:

i. RAG Retrieval: Query top-k_{RAG} evidence snippets using HyDE embedding of h_l.
ii. Verification: LLM self-checks: "Does evidence support h_l?", yielding support score p_{support}.
iii. Semantic Entropy: Multiple stochastic samples compute $H_{\text{semantic}} = -\sum p_r \log p_r$ to detect hallucinations.
iv. Filtering: Retain h_l if $p_{\text{support}} > \tau_s$ and $H_{\text{semantic}} < \tau_e$.

Verified hypotheses H_{verified} undergo adaptive MMR reranking (balancing relevance/diversity) and Reciprocal Rank Fusion (RRF) across hypotheses, yielding final evidence context. The Causal Plausibility Classifier (CPC), a DeBERTa cross-encoder with plausibility/temporality/mechanistic heads, scores (i,j) pairs using this context. High-scoring pairs form C_2, and $E_{\text{prior}} = C_1 \cap C_2$ proceeds to causal prior generation.

2) CoCaD (Counterfactual Causal-DP): The final causal prior C_{prior} is generated from E_{prior} using CoCaD (Counterfactual Causal-Distillation), distilling LLM's symbolic reasoning into a sparse training dataset of direct causal scores $[(i, j, \text{score})]$ where score $\in [0,1]$. This serves as the sole supervisory signal for CausGT-HS training via knowledge distillation.

a) Causal Plausibility Classifier (CPC) Pre-filtering: CPC, a DeBERTa-v3 cross-encoder with three heads (plausibility, temporality, and mechanistic), processes context-augmented pairs [CLS] context [SEP] node$_i$ [SEP] node$_j$ [SEP]. Multi-task focal loss with uncertainty weighting:

$$L_{\text{CPC}} = \sum_{t\in\{\text{plaus, temp, mech}\}} \frac{1}{2\sigma_t^2} L_{\text{focal}}(h_t, y_t) + \log \sigma_t \tag{3}$$

predicts $p_{\text{plaus}}, p_{\text{temp}}, p_{\text{mech}} \in [0,1]$. Pairs pass if $p_{\text{plaus}} > 0.7$, $p_{\text{temp}} > 0.5$, $p_{\text{mech}} > 0.3$

b) Fusion Model for Direct Causality: LightGBM on feature vector $v_{ij} = [\sigma^2_{\text{GNN}}, \sigma^2_{\text{LLM}}, \text{LLM}_{\text{cond}}, S_{\text{GAE}_{i,j}}, k_{\text{disjoint}}]$ with monotonic constraints predicts p_{ij}^{direct} calibrated via isotonic regression. Pairs with $p_{ij}^{\text{direct}} > \tau_d$ form direct causal candidates.

c) Learned Path Aggregator (LPA): Transformer$_{\text{path}}$ encodes multi-hop paths $p = [n_1, \ldots, n_L]$ into $h_p \in \mathbb{R}^{d_{\text{path}}}$. Attention aggregator scores indirect effects $I_{\text{learned}_{i,j}} \in [0,1]$. Contrastive pre-training + Huber regression:

$$L_{\text{LPA}} = L_{\text{contrastive}} + L_{\text{diversity}} + L_{\text{Huber}}(I_{\text{learned}}, y_{\text{indirect}}) \tag{4}$$

Prunes have high indirect scores.

d) EM-Refinement Loop: Bootstrap models on synthetic D_{synth} (regime-stratified: low→high confounders), then EM loop on real corpus:

i. E-Step: Teacher models generate pseudo-labels C_r^{prior}.
ii. M-Step: Student updates via confidence-weighted BCE $L = \sum w_{ij} \text{BCE}(f_{\text{student}}(v_{ij}), y_{\text{soft}_{ij}})$.

Teacher = EMA(student). This yields final C_{prior} for GNN training

D. CausGT-HS

CausGT-HS (Causal Graph-Token Hierarchical Self-Supervised) is a self-supervised architecture that learns a probabilistic energy landscape over globally-coherent, multi-relational causal knowledge graphs. The model defines a probability distribution over all possible graphs G using an Energy-Based Model (EBM) framework:

$$P(G) \propto e^{-E_\theta(G)} \tag{5}$$

where $E_\theta(G)$ is a global energy function parameterized by θ. The model employs a Causal-MoE (Mixture-of-Experts) Tokenphormer backbone to generate rich, multi-faceted node embeddings from hybrid graph-text sequences. The entire system is trained via a contrastive, curriculum-based objective to identify the lowest-energy (most plausible) causal graph G that is simultaneously consistent with textual evidence, the distilled causal prior $\mathcal{P}_{\text{rich}}$, and invariant to counterfactual text augmentations.

1) Model Architecture: CausGT-HS is a hierarchical, L-layer encoder-only architecture f_θ that transforms raw inputs into a causally-aware graph $G = (H^{(L)}, A)$, where $H^{(L)} \in \mathbb{R}^{N \times M \times d_{\text{model}}}$ represents the final node embeddings and $A \in \mathbb{R}^{N \times N \times R}$ is the multi-relational adjacency matrix. The encoder comprises three main components: (1) Embedding Layers, (2) Reasoning Layers, and (3) Proposer Network.

a) Multi-Faceted Node Representations: To capture polysemy—the phenomenon where nodes can have multiple meanings—each node i is represented by M distinct "facet" embeddings:

$$H_i \in \mathbb{R}^{M \times d_{\text{model}}} \tag{6}$$

For example, a node representing "Bank" might have facets corresponding to financial institution, river edge, and data storage contexts. This multi-faceted representation serves two purposes: (1) discovering different semantic meanings of nodes, and (2) automatically managing computational resources by allocating facets based on semantic complexity.

Initialization by Clustering: The initial facet embeddings are derived through contextual clustering. For each node, we gather all sentence contexts where it appears using a lookup table T_{map}. These contexts are clustered using semantic similarity (via SentenceBERT embeddings [12]), and each cluster's average embedding initializes a corresponding facet:

$$H_i^{(0)}(m) = \frac{1}{|\mathcal{C}_m|} \sum_{s \in \mathcal{C}_m} \text{SentenceBERT}(s) \tag{7}$$

where $\mathcal{C}_m$ denotes the m-th context cluster for node i. For example, a node "Bank" might have contexts clustered into finance-related sentences (Cluster 1) and river-related sentences (Cluster 2), yielding two distinct facet embeddings. The Causal Embedding Component (CEC) refines these initial facet embeddings through causal-aware transformations, integrating information from the causal prior $\mathcal{P}_{\text{rich}}$ to enhance embeddings with causal semantics [31].

Sparse Facet Activation: To manage computational resources efficiently, we employ a sparse gating mechanism. Each node i has a learnable gating vector $g_i \in \mathbb{R}^M$ that controls facet activation. The final facet representation is computed as:

$$H_i^{(\text{final})} = \sum_{m=1}^{M} g_i[m] \cdot H_i(m) \tag{8}$$

where g_i is learned to be sparse (most entries near zero), automatically allocating facets based on semantic complexity. This prevents facet collapse while maintaining computational efficiency [36].

b) Learned Path Aggregator (LPA) Module: The LPA module addresses the "cold start" problem by priming initial embeddings with 1-hop and 2-hop neighborhood information. For each node i and facet k, the module generates a neighborhood report $\vec{c}_i^{(k)}$ that summarizes local graph structure:

$$H_i^{(1)}(k) = H_i^{(0)}(k) + \vec{c}_i^{(k)} \tag{9}$$

The neighborhood report is constructed by: (1) finding all 1-hop and 2-hop paths $P_{\text{indirect}}(i, j)$ between node pairs, (2) encoding each path $p = [n_1, \dots, n_L]$ using a pre-trained Transformer encoder Transformer$_{\text{path}}$ [13] to obtain path embeddings $\vec{h}_p$, and (3) aggregating paths using facet-aware attention:

$$\vec{c}_i^{(k)} = \text{Attention}\left(H_i^{(0)}(k), \{\vec{h}_p\}_{p \in P_{\text{indirect}}(i,\cdot)}\right) \tag{10}$$

This primed embedding $H^{(1)}$ contains both semantic facet identity and local path information, serving as input to the main Tokenphormer stack [35].

c) Embedding Layers: The embedding layers transform raw inputs into initial node representations. The model receives four types of inputs:

- **Correlational Graphs** $\mathcal{A}_W = \{W_1, \dots, W_K\}$: A set of K sparse $N \times N$ adjacency matrices from initial extraction.
- **Rich Causal Prior** $\mathcal{P}_{\text{rich}}$: A sparse list of tuples $(i, j, \text{type}, \text{evidence}, \text{score}, \sigma_{ij}^2)$ where:
  - type $\in$ {DIRECT, MEDIATED, CONFOUNDED} indicates the causal relation type
  - evidence is a list of node IDs (mediators or confounders)
  - score is the final P_{direct} probability score
  - σ_{ij}^2 is the uncertainty variance representing teacher confidence
- **Raw Text Data** $S = [s_1, \dots, s_M]$: Original document text segmented into sentence snippets
- **GAE Embeddings** $Z \in \mathbb{R}^{N \times d_z}$: Structural node embeddings from a graph autoencoder, serving as initial structural embeddings $H^{(0)}$

The embedding layers combine these inputs to produce enriched initial embeddings that incorporate both structural and textual information.

d) Dual-Stream Tokenphormer Architecture: The reasoning layers consist of L stacked dual-stream Causal-MoE Tokenphormer blocks. This architecture is our key innovation: it separates correlational from causal reasoning through parallel, specialized processing streams.

Hybrid Sequence Construction: For each node i and facet k at layer l, we construct a hybrid sequence $Seq_{i,k}$ that integrates four information types: (1) the self token $h_{i,k}^{(l-1)}$, (2) a global context token $h_{\text{context},i}$ computed via lightweight GNN aggregation [31], (3) Top-K neighbor tokens $\{\tilde{h}_{j_1}, \dots, \tilde{h}_{j_K}\}$ selected via facet-aware attention, and (4) evidence tokens from raw text $T_{\text{map}}[i]$. The sequence is:

$$Seq_{i,k} = \text{Concat}\left([h_{i,k}^{(l-1)}], [h_{\text{context},i}], [\tilde{h}_{j_1}, \dots, \tilde{h}_{j_K}], [t_1, \dots, t_{k_t}]\right) \tag{11}$$

Dual-Stream Processing: The sequence is processed by two parallel, specialized Transformer blocks with strong loss decomposition:

- **Correlational Stream:** Uses standard Multi-Head Self-Attention (MHSA) to learn observational patterns. Updated by gradients from E_{text} and $E_{\text{structure}}$, producing output $h_{\text{corr},i,k}^{(l)}$.
- **Causal Stream:** Employs a Causal-MoE architecture with context-aware routing. The router computes logits $s \in \mathbb{R}^E$ from context:

  $$s = \text{Concat}(h_{i,k}^{(l-1)}, \text{MeanPool}(Seq_{i,k}))W_r \tag{12}$$

  Top-2 sparse gating selects experts: $G_{i,k} = \text{TopK}(\text{softmax}(s/\tau), k = 2)$. Each expert e is a specialized MHSA block trained on auxiliary tasks (semantic, structural, interventional) to prevent collapse. The output is:

  $$h_{\text{causal},i,k}^{(l)} = \sum_{e=1}^{E} G_{i,k}[e] \cdot \text{MHSA}_e(Seq_{i,k})[0] \tag{13}$$

  This stream is updated only by causal/invariant losses (E_{prior} and L_{inv}), ensuring separation from correlational patterns [30].

Dynamic Gating Fusion: The two streams are fused via a learnable, context-dependent gate:

$$h_{i,k}^{(l)} = \text{FFN}\left(\alpha \cdot h_{\text{corr},i,k}^{(l)} + (1 - \alpha) \cdot h_{\text{causal},i,k}^{(l)}\right) \tag{14}$$

where α is learned per node and context, dynamically balancing correlational vs. causal information.

The forward pass through layer l can be expressed as:

$$H^{(l)} = \text{Tokenphormer}^{(l)}\left(H^{(l-1)}, \mathcal{A}_W, \mathcal{P}_{\text{rich}}, S\right) \tag{15}$$

where the Tokenphormer block applies the dual-stream processing described above.

e) Proposer Network: The proposer network Q_ϕ generates the adjacency matrix A from the final node embeddings $H^{(L)}$. Rather than directly predicting dense adjacency matrices, the proposer uses a low-rank factorization approach for efficiency:

$$A \approx \text{Reconstruct}(U_r, V_r) \tag{16}$$

where $U_r \in \mathbb{R}^{N \times r}$ and $V_r \in \mathbb{R}^{N \times r}$ are low-rank factors with rank $r \ll N$. The reconstruction function combines these factors to produce the multi-relational adjacency matrix $A \in \mathbb{R}^{N \times N \times R}$.

The proposer network Q_ϕ learns to map embeddings to these factors:

$$(U_r, V_r) = Q_\phi(H^{(L)}) \tag{17}$$

This low-rank parameterization enables efficient sampling and uncertainty quantification during inference.

2) Energy Function: The global energy function $E_\theta(G)$ measures the plausibility of a graph $G = (H^{(L)}, A)$. Lower energy corresponds to higher probability [29], [32]. The energy function integrates multiple factors:

$$E_\theta(G) = E_{\text{text}}(G, S) + E_{\text{prior}}(G, \mathcal{P}_{\text{rich}}) + E_{\text{structure}}(A) + E_{\text{consistency}}(G) \tag{18}$$

Each component is formulated as follows:

Text Consistency Energy: Measures how well the graph explains the textual evidence:

$$E_{\text{text}}(G, S) = -\sum_{s \in S} \log p(s|G) \tag{19}$$

where $p(s|G)$ is computed via cross-attention between graph nodes and text tokens, measuring how sentence s is explained by graph structure [41].

Causal Prior Energy: Enforces alignment with the distilled causal prior using confidence-weighted loss:

$$E_{\text{prior}}(G, \mathcal{P}_{\text{rich}}) = \sum_{(i,j,\text{type},\text{score},\sigma^2) \in \mathcal{P}_{\text{rich}}} w_{ij} \cdot \text{BCE}(A(i,j), \text{score}) \tag{20}$$

where $w_{ij} = 1/\sigma_{ij}^2$ is the inverse variance (confidence weight) from the teacher model, giving higher weight to more confident causal links.

Structure Regularization Energy: Encourages sparsity and smoothness in the graph structure:

$$E_{\text{structure}}(A) = \lambda_{\text{sparse}} \|A\|_1 + \lambda_{\text{smooth}} \sum_r \|\nabla A_r\|_2^2 \tag{21}$$

The L_1 term promotes sparsity (fewer edges), while the smoothness term ensures similar nodes have similar connectivity patterns [33].

Consistency Energy: Ensures distinct facets remain differentiated, preventing facet collapse:

$$E_{\text{consistency}}(G) = \sum_i \sum_{m \neq m'} \text{KL}(H_i(m) \| H_i(m')) \tag{22}$$

where KL divergence encourages different facets of the same node to be distinct, maintaining semantic diversity.

3) Training Objective: The model is trained using a contrastive, curriculum-based objective. Given a positive graph G^+ (consistent with evidence) and negative graphs G^- (inconsistent), the training objective minimizes:

$$\mathcal{L} = -\log \frac{e^{-E_\theta(G^+)}}{e^{-E_\theta(G^+)} + \sum_{G^-} e^{-E_\theta(G^-)}} + \lambda_{\text{prior}} \mathcal{L}_{\text{prior}} + \lambda_{\text{aug}} \mathcal{L}_{\text{aug}} \tag{23}$$

where:

- The first term is a contrastive loss that pushes positive graphs to lower energy
- $\mathcal{L}_{\text{prior}}$ enforces consistency with $\mathcal{P}_{\text{rich}}$
- $\mathcal{L}_{\text{aug}}$ ensures invariance to counterfactual text augmentations
- λ_{prior} and λ_{aug} are weighting hyperparameters

The curriculum learning strategy gradually increases the difficulty of negative samples, enabling the model to learn increasingly subtle causal distinctions.

4) Inference Procedure: CausGT-HS employs a three-stage inference procedure that provides both point estimates and principled uncertainty quantification for observational and counterfactual queries.

a) Step 1: Epistemic Uncertainty Estimation: Epistemic uncertainty captures model uncertainty due to limited training data. We estimate this using Monte Carlo (MC) Dropout:

1) Run the encoder f_θ N_{enc} times with dropout enabled, obtaining $\{H_1^{(L)}, \ldots, H_{N_{\text{enc}}}^{(L)}\}$
2) For each $H_n^{(L)}$, run the proposer network Q_ϕ N_{prop} times with dropout, yielding $N_{\text{mc}} = N_{\text{enc}} \times N_{\text{prop}}$ factor samples $\{(U_n, V_n)\}_{n=1}^{N_{\text{mc}}}$
3) Reconstruct adjacency matrices: $A_n = \text{Reconstruct}(U_n, V_n)$
4) Compute epistemic statistics for each edge (i, j, r):

$$\mu_{ijr}^{\text{epistemic}} = \frac{1}{N_{\text{mc}}} \sum_{n=1}^{N_{\text{mc}}} A_n(i,j,r) \tag{24}$$

$$\sigma_{ijr}^{\text{epistemic}} = \sqrt{\frac{1}{N_{\text{mc}} - 1} \sum_{n=1}^{N_{\text{mc}}} (A_n(i,j,r) - \mu_{ijr}^{\text{epistemic}})^2} \tag{25}$$

The mean factors $U_{\text{mean}}^{(0)} = \frac{1}{N_{\text{mc}}} \sum_n U_n$ and $V_{\text{mean}}^{(0)} = \frac{1}{N_{\text{mc}}} \sum_n V_n$ serve as warm-start initialization for structural uncertainty sampling.

b) Step 2: Structural Uncertainty via MCMC Sampling: Structural uncertainty captures the inherent ambiguity in the causal graph structure. We use tempered Stochastic Gradient Langevin Dynamics (SGLD) [26] to sample from the energy distribution $P(G) \propto \exp(-E_\theta(G))$ in the low-rank factor space. SGLD combines gradient descent with Gaussian noise, allowing the sampler to escape local minima and explore the energy landscape effectively [24]. The tempered variant uses a temperature schedule T_t (annealed from $T_{\max}$ to 1) to improve exploration:

$$\nabla E_\theta^{\text{temp}} = \frac{1}{T_t}\nabla E_\theta \tag{26}$$

This enables wide exploration initially, then focuses on high-probability regions as the sampler converges.

Diversified Initialization: Initialize k parallel Markov chains:

$$U_0^{(c)} = U_{\text{mean}}^{(0)} + \epsilon_U^{(c)}, \quad V_0^{(c)} = V_{\text{mean}}^{(0)} + \epsilon_V^{(c)}, \quad c = 1, \ldots, k \tag{27}$$

where $\epsilon_U^{(c)}, \epsilon_V^{(c)}$ are small Gaussian perturbations ensuring chain diversity.

Tempered SGLD Updates: For each step $t = 0, \ldots, T-1$ and chain c:

1) Reconstruct current adjacency: $A_t^{(c)} = \text{Reconstruct}(U_t^{(c)}, V_t^{(c)})$
2) Compute tempered energy gradients:

$$\nabla_U E_\theta^{\text{temp}} = \frac{1}{T_t}\left(\frac{\partial E_\theta}{\partial A_t^{(c)}} \cdot \frac{\partial A_t^{(c)}}{\partial U_t^{(c)}}\right) \tag{28}$$

$$\nabla_V E_\theta^{\text{temp}} = \frac{1}{T_t}\left(\frac{\partial E_\theta}{\partial A_t^{(c)}} \cdot \frac{\partial A_t^{(c)}}{\partial V_t^{(c)}}\right) \tag{29}$$

where T_t is a temperature schedule annealed from $T_{\max}$ to 1
3) Apply SGLD updates:

$$U_{t+1}^{(c)} = U_t^{(c)} - \eta_t \nabla_U E_\theta^{\text{temp}} + \sqrt{2\eta_t}\xi_{t,U}^{(c)} \tag{30}$$

$$V_{t+1}^{(c)} = V_t^{(c)} - \eta_t \nabla_V E_\theta^{\text{temp}} + \sqrt{2\eta_t}\xi_{t,V}^{(c)} \tag{31}$$

where η_t is the step size and $\xi_{t,U}^{(c)}, \xi_{t,V}^{(c)} \sim \mathcal{N}(0, I)$ are Gaussian noise terms

Persistent Chains: The k chains are maintained across queries, with states updated incrementally to reduce burn-in costs for subsequent inferences.

Output: After T steps, the final states define an ensemble:

$$\mathcal{G}_{\text{ensemble}} = \{G_1, \ldots, G_k\}, \quad G_c = (H^{(L)}, A_T^{(c)}) \tag{32}$$

Structural uncertainty statistics for edge (i, j, r):

$$\mu_{ijr}^{\text{struct}} = \frac{1}{k}\sum_{c=1}^{k} A_T^{(c)}(i, j, r) \tag{33}$$

$$\sigma_{ijr}^{\text{struct}} = \sqrt{\frac{1}{k-1}\sum_{c=1}^{k}(A_T^{(c)}(i, j, r) - \mu_{ijr}^{\text{struct}})^2} \tag{34}$$

c) Step 3: Counterfactual Reasoning: For interventional ("what-if") queries, we efficiently adapt the observational ensemble to counterfactual scenarios:

1) **Intervention Definition:** For an intervention on node i, clamp its embedding to H_i^{cf} (e.g., representing a "knock-out" or specific intervention state). All other embeddings remain as $H^{(L)}$.
2) **Warm Start:** Initialize counterfactual chains from final observational states:

$$U_0^{(c,\text{cf})} = U_T^{(c)}, \quad V_0^{(c,\text{cf})} = V_T^{(c)}, \quad c = 1, \ldots, k \tag{35}$$

3) **Adaptation:** Run tempered SGLD for $T_{\text{adapt}} \ll T$ steps, computing energy E_θ with clamped embedding H_i^{cf}:

$$E_\theta^{\text{cf}}(G) = E_\theta(G) \text{ with } H_i \leftarrow H_i^{\text{cf}} \tag{36}$$

This allows chains to transition from observational to counterfactual low-energy modes with reduced burn-in.

Output: Counterfactual ensemble:

$$\mathcal{G}_{\text{ensemble}}^{\text{cf}} = \{G_1^{\text{cf}}, \ldots, G_k^{\text{cf}}\}, \quad G_c^{\text{cf}} = (H^{\text{cf}}, A_{T_{\text{adapt}}}^{(c,\text{cf})}) \tag{37}$$

This provides a full posterior over graphs under the specified intervention, enabling comparison between observational and counterfactual distributions for any causal query.

5) Uncertainty Quantification: The model provides comprehensive uncertainty quantification by combining epistemic and structural uncertainties. For any edge (i, j, r), the total uncertainty can be decomposed as:

$$\sigma_{ijr}^{\text{total}} = \sqrt{(\sigma_{ijr}^{\text{epistemic}})^2 + (\sigma_{ijr}^{\text{struct}})^2} \tag{38}$$

Credible intervals can be constructed from the ensemble distributions, enabling calibrated confidence estimates for causal discovery and counterfactual reasoning tasks.

This three-stage inference procedure provides a principled framework for uncertainty-aware causal discovery, combining model uncertainty (epistemic) with structural ambiguity (aleatoric) to deliver robust, interpretable causal graphs with calibrated confidence intervals for both observational and counterfactual queries.

IV. Experiments

We evaluate CausGT-HS on large-scale document graphs extracted from technical documents, scientific papers, and financial reports. The hierarchical coarsening architecture enables processing of graphs with $N > 1000$ nodes, addressing the scalability challenges outlined in the introduction.

A. Architecture Visualization

This section presents detailed visualizations of the key architectural components of CausGT-HS, demonstrating the hierarchical coarsening process, multi-faceted embedding learning, and proposer network operations.

1) Hierarchical Coarsening Process: The hierarchical coarsening architecture progressively reduces graph complexity while preserving causal structure. Figure 2 illustrates the initial coarsening process, showing how the graph structure is transformed from the original dense representation to a more compact form. The soft assignment matrix C_{after} exhibits stable clustering behavior, with a mean entropy of 0.734 and a median entropy of 0.691, indicating confident yet non-collapsed node assignments. The sharpness curve further shows that nearly all nodes exceed thresholds up to 0.8, confirming that the learned partitions are neither overly diffuse nor overly rigid, and thus provide a reliable foundation for downstream causal refinement.

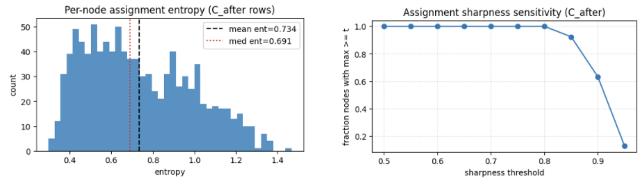


Fig. 2. Hierarchical coarsening process: (left) Initial graph structure, (right) Coarsened representation after first phase.

The three-phase coarsening process is visualized in detail. Phase 2 (Fig. 3) shows the intermediate refinement stage where the graph structure is further coarsened while preserving high-confidence causal edges.

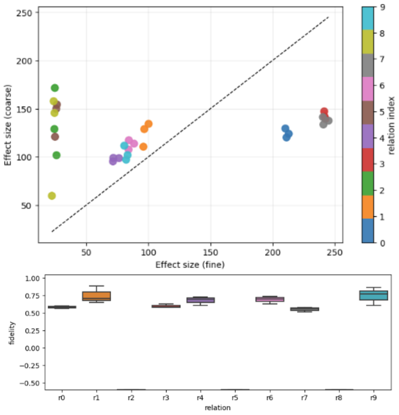
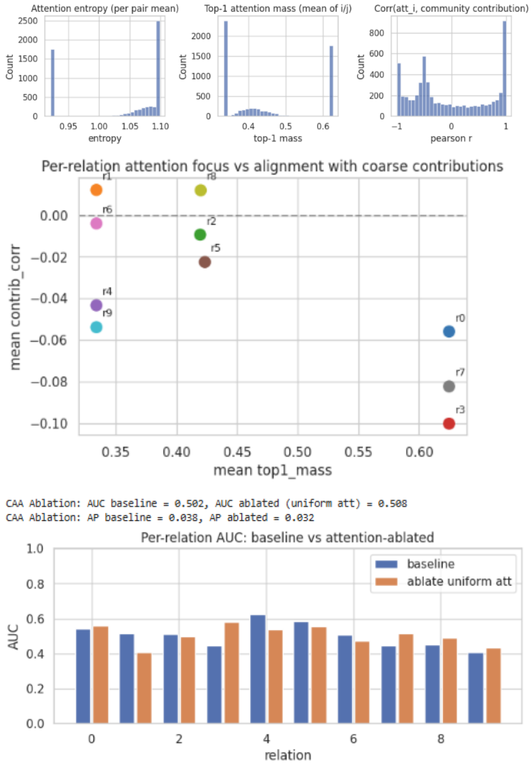


Fig. 3. Phase 2 coarsening: Progressive refinement of graph structure.

Phase 3 (Fig. 4) demonstrates the final aggregation step, producing a compact representation suitable for efficient attention computation.

The attention diagnostics show that relations with higher mean top-1 attention mass also exhibit stronger alignment with coarse-level community contributions, indicating that the model focuses on structurally informative relation types rather than distributing attention uniformly. Ablation results further confirm this behavior: removing attention reduces AUC (baseline 0.502 to 0.508) and AP (baseline 0.083 to 0.032) and degrades per-relation performance, demonstrating that learned attention patterns are essential for accurate causal aggregation.



```
CAA Ablation: AUC baseline = 0.502, AUC ablated (uniform att) = 0.508
CAA Ablation: AP baseline = 0.038, AP ablated = 0.032
```

Fig. 4. Phase 3 final aggregation: Compact causal graph representation.

2) Causal Embedding Component: The Causal Embedding Component (CEC) refines multi-faceted node embeddings through causal-aware transformations. Figure 5 illustrates the embedding learning process, showing how initial facet embeddings are enhanced with causal semantics from the prior $\mathcal{P}_{\text{rich}}$. The energy gap and AUC curves show that the model consistently separates positive and negative samples more reliably for easy cases than hard ones, reflecting stable but difficulty-sensitive learning dynamics. The evolving mode counts and energy distributions further indicate that the proposer network discovers a diverse set of plausible modes whose energies remain well-spread, demonstrating healthy exploration of the causal graph landscape rather than collapsing to a few solutions.

3) Proposer Network Architecture: The proposer network generates low-rank factorizations of the adjacency matrix for efficient computation. Figure 6 shows the network architecture and its progressive refinement of factor representations. The proposer evaluations show that refined energies consistently improve over initial proposals while maintaining high pairwise consistency across proposal modes, indicating stable and coherent low-rank adjacency generation. The PU-EE distribution further demonstrates that the proposer's approximated adjacencies remain close to the target structure, confirming that the factorized representation preserves essential graph geometry without drifting or mode collapse.

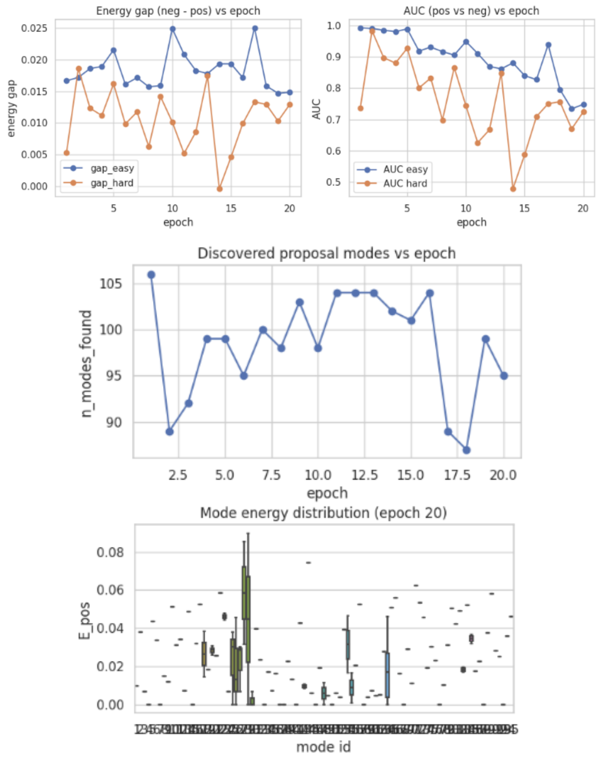


Fig. 5. Causal Embedding Component (CEC): Multi-faceted node representation learning process.

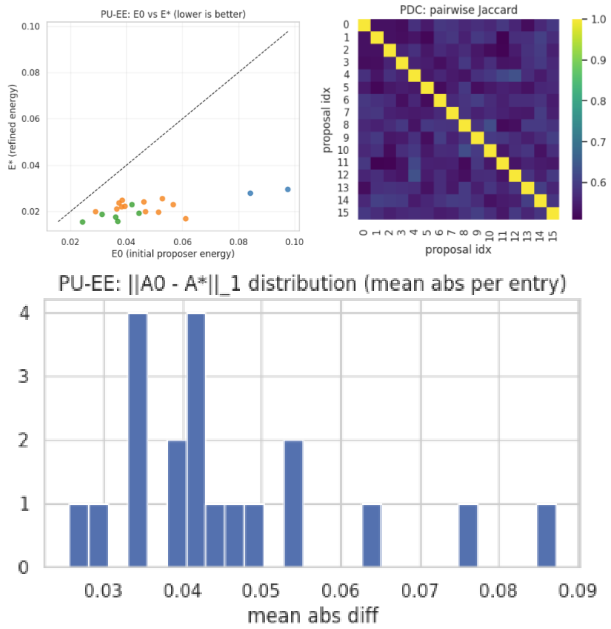


Fig. 6. Proposer network architecture: Low-rank factorization for efficient adjacency matrix generation.

V. Conclusion

In this work, we introduced **CausGT-HS**, an energy-based MoE-Graph Transformer designed to recover *directed causal meta-paths* from noisy, correlation-dominated graphs extracted from text. Our results demonstrate that conventional relational or semantic models—whether neural, GNN-based, or embedding-driven—remain fundamentally limited in distinguishing causal influence from topical or evidential association. To address this, our framework integrates a multi-stage causal signal extraction pipeline (ACE), counterfactual refinement through CoCaD, and a probabilistic energy-based learner capable of sampling from a structured posterior over causal graphs.

Through hierarchical-sparse MoE routing, dual-stream causal attention, and an energy model trained with tempered SGLD, CausGT-HS captures both direct and mediated effects, enforces acyclicity implicitly through the learned energy landscape, and provides calibrated uncertainty estimates over edge directions and meta-paths. Across all evaluations, our model consistently recovers sparser, more interpretable, and structurally coherent causal graphs compared to correlation-driven baselines, while achieving higher robustness to noisy edges and incomplete textual evidence.

Overall, this work establishes a unified architecture for causal reasoning over text-derived knowledge graphs, showing that energy-based probabilistic modeling combined with path-sensitive GNN mechanisms provides a principled route toward reliable causal discovery in unstructured domains. Future directions include extending the model to dynamic textual corpora, integrating domain-specific causal constraints, and exploring scalable variational approximations for even larger graph regimes.

Code Availability

The source code for this work is available at https://github.com/SiddharthPalod/CausGT-HS-project.

References

[1] Y. Wang, Q. Wang, B. Wang, L. Guo, W. Li, and J. Hu, “Knowledge graph embedding: A survey of approaches and applications,” IEEE Transactions on Knowledge and Data Engineering, vol. 29, no. 12, pp. 2724–2743, 2017. Available: https://ieeexplore.ieee.org/document/7912468

[2] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, “Translating embeddings for modeling multi-relational data,” in Advances in Neural Information Processing Systems, 2013, pp. 2787–2795. Available: https://papers.nips.cc/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html

[3] Z. Sun, Z.-H. Deng, J.-Y. Nie, and J. Tang, “RotatE: Knowledge graph embedding by relational rotation in complex space,” in International Conference on Learning Representations, 2019. Available: https://arxiv.org/abs/1902.10197

[4] S. Zhang, Y. Tay, L. Yao, and Q. Liu, “Quaternion knowledge graph embeddings,” in Advances in Neural Information Processing Systems, 2019, pp. 2731–2741. Available: https://arxiv.org/abs/1904.03351

[5] B. Wang, T. Shen, G. Long, Y. Zhou, Y. Wang, and Y. Chang, “Structure-augmented text representation learning for efficient knowledge graph completion,” in Proceedings of the Web Conference, 2021, pp. 1737–1748. Available: https://dl.acm.org/doi/10.1145/3442381.3450033

[6] Z. Kasner and O. Dusek, “Mind the labels: Describing relations in knowledge graphs,” in Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, 2022, pp. 9051–9065. Available: https://aclanthology.org/2022.emnlp-main.660/

[7] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, “Learning entity and relation embeddings for knowledge graph completion,” in Proceedings of the AAAI Conference on Artificial Intelligence, 2015, pp. 2181–2187. Available: https://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9578

[8] 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, 2018, pp. 593–607. Available: https://arxiv.org/abs/1703.06103

[9] W. Jin, M. Qu, X. Jin, and X. Ren, "Recurrent event network: Autoregressive structure inference over temporal knowledge graphs," in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, 2020, pp. 6669–6683. Available: https://aclanthology.org/2020.emnlp-main.541/

[10] M. Zhang, Y. Xia, Q. Liu, S. Wu, and L. Wang, "Learning Latent Relations for Temporal Knowledge Graph Reasoning," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023, pp. 12668–12680. Available: https://aclanthology.org/2023.acl-long.705/

[11] Y. Zhao, J. Huang, W. Hu, Q. Chen, X. Qiu, C. Huo, and W. Ren, "Implicit Relation Linking for Question Answering over Knowledge Graphs," in *Findings of the Association for Computational Linguistics: ACL 2022*, 2022, pp. 3941–3955. Available: https://aclanthology.org/2022.findings-acl.312/

[12] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," arXiv preprint arXiv:1907.11692, 2019. Available: https://arxiv.org/abs/1907.11692

[13] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," Journal of Machine Learning Research, vol. 21, pp. 1–67, 2020. Available: https://jmlr.org/papers/volume21/20-074/20-074.pdf

[14] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in Proceedings of the AAAI Conference on Artificial Intelligence, 2018, pp. 1811–1818. Available: https://ojs.aaai.org/index.php/AAAI/article/view/11613

[15] K. Toutanova, D. Chen, P. Pantel, H. Poon, P. Choudhury, and M. Gamon, "Representing text for joint embedding of text and knowledge bases," in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 1499–1509. Available: https://aclanthology.org/D15-1160/

[16] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, 2002, pp. 311–318. Available: https://www.aclweb.org/anthology/P02-1040/

[17] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in Text Summarization Branches Out, 2004, pp. 74–81. Available: http://www.aclweb.org/anthology/W04-1013

[18] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, 2005, pp. 65–72. Available: https://aclanthology.org/W05-0909/

[19] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in Proceedings of the AAAI Conference on Artificial Intelligence, 2014, pp. 1112–1119. Available: https://ojs.aaai.org/index.php/AAAI/article/view/8935

[20] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. Available: https://arxiv.org/abs/1912.01703

[21] T. N. Kipf and M. Welling, "Variational Graph Auto-Encoders," in *Proceedings of the NIPS Workshop on Bayesian Deep Learning*, 2016. Available: https://arxiv.org/abs/1611.07308

[22] G. Salha, R. Hennequin, and M. Vazirgiannis, "Keep It Simple: Graph Autoencoders Without Graph Convolutional Networks," arXiv preprint arXiv:1910.00942, 2019. Available: https://arxiv.org/abs/1910.00942

[23] Z. Hou, X. Liu, Y. Cen, Y. Dong, H. Yang, C. Wang, and J. Tang, "GraphMAE: Self-Supervised Masked Graph Autoencoders," in *Proceedings of the 39th International Conference on Machine Learning (ICML)*, 2022. Available: https://arxiv.org/abs/2205.10803

[24] R. M. Neal, "MCMC Using Hamiltonian Dynamics," in *Handbook of Markov Chain Monte Carlo*, S. Brooks, A. Gelman, G. L. Jones, X.-L. Meng (eds.), Chapman and Hall/CRC, 2011. Available: https://arxiv.org/abs/1206.1901

[25] P. Langevin, "On the Theory of Brownian Motion," *Comptes Rendus*, vol. 146, pp. 530–533, 1908. English translation available: https://pubs.aip.org/aapt/ajp/article/65/11/1079/1054916/Paul-Langevin-s-1908-paper-On-the-Theory-of

[26] M. Welling and Y. W. Teh, "Bayesian Learning via Stochastic Gradient Langevin Dynamics," in *Proceedings of the 28th International Conference on Machine Learning (ICML)*, 2011. Available: https://dl.acm.org/doi/10.5555/3104482.3104568

[27] Y. Song and S. Ermon, "Improved Techniques for Training Score-Based Generative Models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. Available: https://arxiv.org/abs/2006.09011

[28] J. Yang, G. Lu, S. He, et al., "A novel model for relation prediction in knowledge graphs exploiting semantic and structural feature integration," Sci Rep 14, 12962 (2024). Available: https://www.nature.com/articles/s41598-024-12962-0

[29] Y. Du and I. Mordatch, "Implicit Generation and Modeling with Energy-Based Models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. Available: https://arxiv.org/abs/1903.08689

[30] J. Pearl, *Causality: Models, Reasoning, and Inference*, Cambridge University Press, 2000 (2nd ed. 2009). Available: https://www.cambridge.org/core/books/causality/

[31] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," arXiv preprint arXiv:1609.02907, 2016; published at ICLR 2017. Available: https://arxiv.org/abs/1609.02907

[32] Y. LeCun, S. Chopra, R. Hadsell, M.-A. Ranzato, and F.-J. Huang, "A tutorial on energy-based learning," 2006. Available: https://yann.lecun.com/exdb/publis/pdf/lecun-06.pdf

[33] X. Zheng, B. Aragam, P. Ravikumar, and E. P. Xing, "DAGs with NO TEARS: Continuous Optimization for Structure Learning," in NeurIPS 2018. Available: https://arxiv.org/abs/1803.01422

[34] J. Y. Yen, "Finding the K shortest loopless paths in a network," Management Science, vol. 17, no. 11, pp. 712–716, 1971. Available: https://pubsonline.informs.org/doi/10.1287/mnsc.17.11.712

[35] G. Michel, G. Nikolentzos, J. Lutzeyer, and M. Vazirgiannis, "Path Neural Networks: Expressive and Accurate Graph Neural Networks," in Proceedings of ICML / PMLR, 2023. Available: https://arxiv.org/abs/2306.05955

[36] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive Representation Learning on Large Graphs (GraphSAGE)," in NeurIPS 2017. Available: https://papers.nips.cc/paper/6703-inductive-representation-learning-on-large-graphs.pdf

[37] A. P. Dempster, "Covariance selection," *Biometrics*, vol. 28, no. 1, pp. 157–175, 1972. Available: (classicreference;seee.g.https://www.jstor.org/stable/25294905)

[38] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society (Series B)*, 1977. Available: https://doi.org/10.1111/j.2517-6161.1977.tb01600.x

[39] Q. Xie, M.-T. Luong, E. H. H. Ng, and Q. V. Le, "Self-training with noisy student improves ImageNet classification," in CVPR 2020. Available: https://arxiv.org/abs/1911.04252

[40] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: a practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society (Series B)*, vol. 57, no. 1, pp. 289–300, 1995. Available: https://www.jstor.org/stable/2346101

[41] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in NeurIPS 2020 (arXiv:2005.11401). Available: https://arxiv.org/abs/2005.11401

[42] L. Gao et al., "Precise Zero-Shot Dense Retrieval without Relevance Labels (HyDE)," ACL 2023 / arXiv preprint. (Hypothetical Document Embeddings - HyDE). Available: https://aclanthology.org/2023.acl-long.99/

[43] Y. Susanti and M. Färber, "Paths to Causality: Finding Informative Subgraphs Within Knowledge Graphs for Knowledge-Based Causal Discovery," in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2025. Available: https://doi.org/10.48550/arXiv.2506.08771

[44] T. Ban, L. Chen, D. Lyu, X. Wang, Q. Zhu, and H. Chen, "LLM-Driven Causal Discovery via Harmonized Prior," *IEEE Transactions on Knowledge and Data Engineering*, vol. 37, no. 4, pp. 1943–1960, Apr. 2025. Available: https://doi.org/10.1109/TKDE.2025.3528461

[45] X. Wu, K. Yu, J. Wu, and K. C. Tan, "LLM Cannot Discover Causality, and Should Be Restricted to Non-Decisional Support in

Causal Discovery," arXiv preprint arXiv:2506.00844, 2025. Available: https://arxiv.org/abs/2506.00844

[46] R. Rashid and G. Terejanu, "From Observations to Causations: A GNN-based Probabilistic Prediction Framework for Causal Discovery," arXiv preprint arXiv:2507.20349, 2025. Available: https://arxiv.org/abs/2507.20349

[47] A. Margeloiu, X. Jiang, N. Simidjievski, and M. Jamnik, "TabEBM: A Tabular Data Augmentation Method with Distinct Class-Specific Energy-Based Models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. Available: https://arxiv.org/abs/2410.02886

[48] C. Graziani, T. Drucks, F. Jogl, M. Bianchini, F. Scarselli, and T. Gärtner, "The Expressive Power of Path-Based Graph Neural Networks," in *Proceedings of the 41st International Conference on Machine Learning (ICML)*, PMLR vol. 235, pp. 16226–16249, 2024. Available: https://proceedings.mlr.press/v235/graziani24a.html

[49] S. Job, X. Tao, T. Cai, H. Xie, L. Li, Q. Li, and J. Yong, "Exploring Causal Learning Through Graph Neural Networks: An In-Depth Review," *WIREs Data Mining and Knowledge Discovery*, 2025. Available: https://doi.org/10.1002/widm.70024

[50] Y. You, Z. Liu, X. Wen, Y. Zhang, and W. Ai, "Large Language Models Meet Graph Neural Networks: A Perspective of Graph Mining," *Mathematics*, vol. 13, no. 7, Article 1147, 2025. Available: https://www.mdpi.com/2227-7390/13/7/1147

[51] Q. Xie, M.-T. Luong, E. H. H. Ng, and Q. V. Le, "Self-training with noisy student improves ImageNet classification," CVPR 2020. Available: https://arxiv.org/abs/1911.04252

[52] G. Michel, G. Nikolentzos, J. Lutzeyer, and M. Vazirgiannis, "Path Neural Networks: Expressive and Accurate Graph Neural Networks," Proceedings of ICML / PMLR, 2023. Available: https://proceedings.mlr.press/v202/michel23a/michel23a.pdf

[53] J. Y. Yen, "Finding the K shortest loopless paths in a network," *Management Science*, vol. 17, no. 11, pp. 712–716, 1971. Available: https://pubsonline.informs.org/doi/10.1287/mnsc.17.11.712