Tree-of-Thought and Perception-Oriented Knowledge Graph Generation

Rahul Krishna Gaddam

September 23, 2025

Abstract

This report presents a rigorous framework for knowledge graph (KG) generation that combines Tree-of-Thought (ToT) sequential reasoning with perception-oriented, multi-instance embeddings. The methodology enables multi-view KG generation, semantic projection invariance, temporal and spatial drift analysis, and canonical embedding merging. We formalize graph generation, embedding projection, perception-specific variations, and multi-instance representation with mathematical precision.

1 Tree-of-Thought KG Generation

1.1 Sequential Graph Reasoning

Let G_t denote a partially constructed graph at time t. Tree-of-Thought reasoning models KG construction as a tree of candidate paths:

$$T_G = \{\tau_1, \tau_2, \ldots, \tau_k\},$$

where each branch τ_i is a sequence of graph construction steps:

$$\tau_i = (s_0, s_1, \ldots, s_{L_i}),$$

with s_j representing node addition, relation creation, or attribute update. The root s_0 corresponds to the initial graph, leaves correspond to candidate graphs $G(\tau_i)$.

1.2 Dynamic Schema and Multi-View Embeddings

Let the schema at time t be

$$S_t = \{v_i, R_{ij}\}, \quad i, j \in [1, M],$$

where v_i are node types and R_{ij} allowed relations.

Each branch embedding is generated by:

$$e_{\tau_i} = f_{\mathrm{ToT}}(\tau_i) \in \mathbb{R}^d,$$

capturing semantic focus, spatial drift (orthogonal subspaces), and temporal drift.

1.3 Merging Candidate Graphs

Two approaches exist:

- **Canonical merging:**
$$e_G = \sum_i w_i P_S e_{\tau_i}, \quad \sum_i w_i = 1, \quad w_i \geq 0,$$
with w_i attention weights from path likelihood.
- **Independent storage:** store each e_{τ_i} separately and resolve multiplicity via clustering of $P_S e_{\tau_i}$.

1.4 Temporal and Semantic Drift

Define drift between consecutive versions:

$$\Delta e(t) = e_G(t) - e_G(t-1) = P_S \Delta e(t) + (I - P_S)\Delta e(t),$$

where $P_S \Delta e(t)$ captures semantic evolution, and $(I - P_S)\Delta e(t)$ captures residual changes.

Temporal weighting integrates historical relevance:

$$e_{G,\text{merged}} = \sum_t K(t)\, e_G(t).$$

1.5 Spatial Drift via Subspaces

For branch embeddings:

$$\text{Semantic projection: } P_S e_{\tau_i}, \quad \text{Residual: } (I - P_S) e_{\tau_i}.$$

1.6 Loss Function for ToT KG Generation

$$L_{\text{ToT}} = \sum_i \|P_S e_{\tau_i} - e_{\text{canonical}}\|^2 + \lambda \sum_i \|(I - P_S) e_{\tau_i}\|^2 - \mu \sum_{i,j} \text{sim}(e_{\tau_i}, e_{\tau_j}).$$

2 Perception-Oriented Multi-Instance KG Generation

2.1 Perception Functions

Let

$$P = \{\pi_1, \pi_2, \ldots, \pi_K\},$$

where $\pi_k : S_t \to G_t(\pi_k)$ maps dynamic schema and available data to a candidate graph.

A ToT branch explores sequential perceptions:

$$\tau_i = (\pi_{k_1}, \pi_{k_2}, \ldots, \pi_{k_L}) \implies G_t(\tau_i) = \pi_{k_L} \circ \cdots \circ \pi_{k_1}(S_t),$$

encoding multi-step reasoning to generate multi-view KG candidates.

2.2 Embedding Multi-Perception Graphs

For each perception-generated graph:

$$e_t(\pi_k) = f_{\text{enc}}(G_t(\pi_k)) \in \mathbb{R}^d,$$

with semantic projection $P_S e_t(\pi_k)$ preserving identity-critical components. Residual $(I - P_S) e_t(\pi_k)$ captures perception-specific variations.

Multiple embeddings $\{e_t(\pi_k)\}_{k=1}^K$ correspond to different perspectives of the same entity.

2.3 Semantic Consistency Across Perceptions

To enforce semantic identity invariance:

$$\text{proj}_S(e_t(\pi_k)) \approx \text{proj}_S(e_t(\pi_j)), \quad \forall k, j.$$

Residuals $(I - P_S) e_t(\pi_k)$ encode perception-specific differences, retained for multi-instance storage.

2.4 Multi-Instance Representation

For unique entity u at time t:

$$G_{u,t} = \{G_t(\pi_1), \ldots, G_t(\pi_K)\}, \quad E_{u,t} = \{e_t(\pi_1), \ldots, e_t(\pi_K)\}.$$

Canonical representation via attention-weighted merge:

$$e_{u,\text{canonical}} = \sum_{k=1}^{K} w_k P_S e_t(\pi_k), \quad \sum_k w_k = 1.$$

Residuals $(I - P_S)e_t(\pi_k)$ are stored independently, preserving multiple graph instances per entity.

2.5 Temporal and Perception Drift

Sequential updates:

$$\Delta e_t(\pi_k) = e_t(\pi_k) - e_{t-1}(\pi_k) = P_S \Delta e_t(\pi_k) + (I - P_S)\Delta e_t(\pi_k),$$

where semantic drift $P_S \Delta e_t(\pi_k)$ captures core identity evolution, and residual $(I - P_S)\Delta e_t(\pi_k)$ captures perception-specific variations.

2.6 Integration into Vector-Semantic KG Storage

Each perception graph $G_t(\pi_k)$ generates embedding $e_t(\pi_k)$. GraphDB stores both node/relation structure and embeddings. Semantic retrieval uses $P_S e_t(\pi_k)$ to ensure consistent identity retrieval, while residuals preserve perception multiplicity.

3 References

1. Yao et al., *Tree-of-Thought Reasoning for Knowledge Graph Construction*, 2024.
2. Zeng et al., *Multi-View Graph Representation Learning*, 2022.
3. Huang et al., *Dynamic Schema Graphs and Temporal Knowledge Graphs*, 2021.
4. Li et al., *Semantic Drift in Knowledge Graph Embeddings*, 2020.