Sparsity and Noise: Where Knowledge Graph Embeddings Fall Short

***by* Bhagyasha Patil**

***on* 26th November, 2024**

Table of Contents

1. Background

Knowledge Graphs

- Structured triples: (subject, predicate, object).
- Applications: Question answering, decision support, semantic search.

Embeddings

- Represent entities/relations in low-dimensional vectors.
- Purpose: Link prediction, KG completion, error correction.

Challenges

- Sparsity: Incomplete data for many entities/relations.
- Noise: Errors from automatic extraction.

2. Dataset Overview

- Freebase: 1B triples, 124M entities, highly curated.
- WordNet: 380K triples, small but precise.
- NELL: Noisy extractions, precision ~35–85%.
- FB15K: Subset of Freebase.
- WN18: Subset of WordNet.
- NELL165: Noisy, real-world data.

3. Key Concepts

Sparsity

- Lack of observations per entity/relation → Hard to train embeddings.

Reliability

- High precision: Curated datasets (e.g., Freebase).
- Low precision: Extracted datasets (e.g., NELL).

Diversity

- Distribution of facts across entities/relations.

4. Experimental Setup

Embedding Methods: TransE, TransH, HolE, STransE

Metrics

- AUPRC: Area under precision-recall curve.
- Hits@10: Top 10 ranked triples.

5. Results

- Sparse and noisy data → Poor embedding performance.
- Sparsity hurts performance → Dense data is key.
- Noise harms embeddings → Clean triples improve results.
- Trade-off: Low-noise triples help; high-noise triples harm.

6. Recommendations

- Combine embeddings with probabilistic reasoning.
- Use confidence scores for optimization.
- Explore open-world embedding models.

4. Conclusion

- Embeddings struggle with sparse/noisy real-world data.
- Dense, high-quality datasets are essential.
- Future work: Open-world assumptions and hybrid models.