KGMark: A Diffusion Watermark for Knowledge Graphs

Hongrui Peng, Haolang Lu, Yuanlong Yu, Weiye Fu, Kun Wang, Guoshun Nan

Beijing University of Posts and Telecommunications

Nanyang Technological University

ICML 2025

Problem: Securing AI-Generated Knowledge Graphs

Context:

- Knowledge Graphs (KGs) are crucial for many AI applications, including semantic search and recommendation systems.
- AI models can now generate high-quality synthetic KGs.

Challenges:

- **Integrity & Bias:** Synthetic KGs can embed biases or misleading information, and are vulnerable to malicious alterations.
- **Intellectual Property:** Presenting synthetic graphs as original work can violate IP rights, undermining trust.
- **Technical Gap:** Existing watermarking fails on dynamic graphs due to spatial and temporal variations.

Goal:

- To develop the first watermarking framework for KGs that generates **robust**, **detectable**, and **transparent** fingerprints.

Motivation and State-of-the-Art

Limitations of Prior Work:

- Conventional watermarking lacks robustness against the spatial-temporal variations of dynamic KGs.
- The heterogeneity of KGs requires embedding at the embedding level to balance fidelity and resilience.
- No existing method effectively handles unique graph attacks like isomorphism and structural perturbations.

Open Questions Addressed by this Paper:

- Can we design a robust watermark for KGEs against structural and temporal changes?
- Is it possible to embed a watermark with minimal impact on the KG's quality and downstream task performance?
- Can we create a secure and computationally feasible algorithm?

KGMark is the first framework to systematically solve these issues for KGs using a novel diffusion-based approach.

KGMark: The Three Pillars

KGMark is a comprehensive framework designed to satisfy the three essential properties of an effective watermark.

Transparency

- Minimal impact on the KG's usability.
- Preserves graph structure and semantics.

Detectability

- Accurate identification of the watermark's presence.
- High detection rates with confidence.

Robustness

- Resilience to attacks and modifications.
- Withstands post-editing and structural changes.

KGMark: The Three Pillars

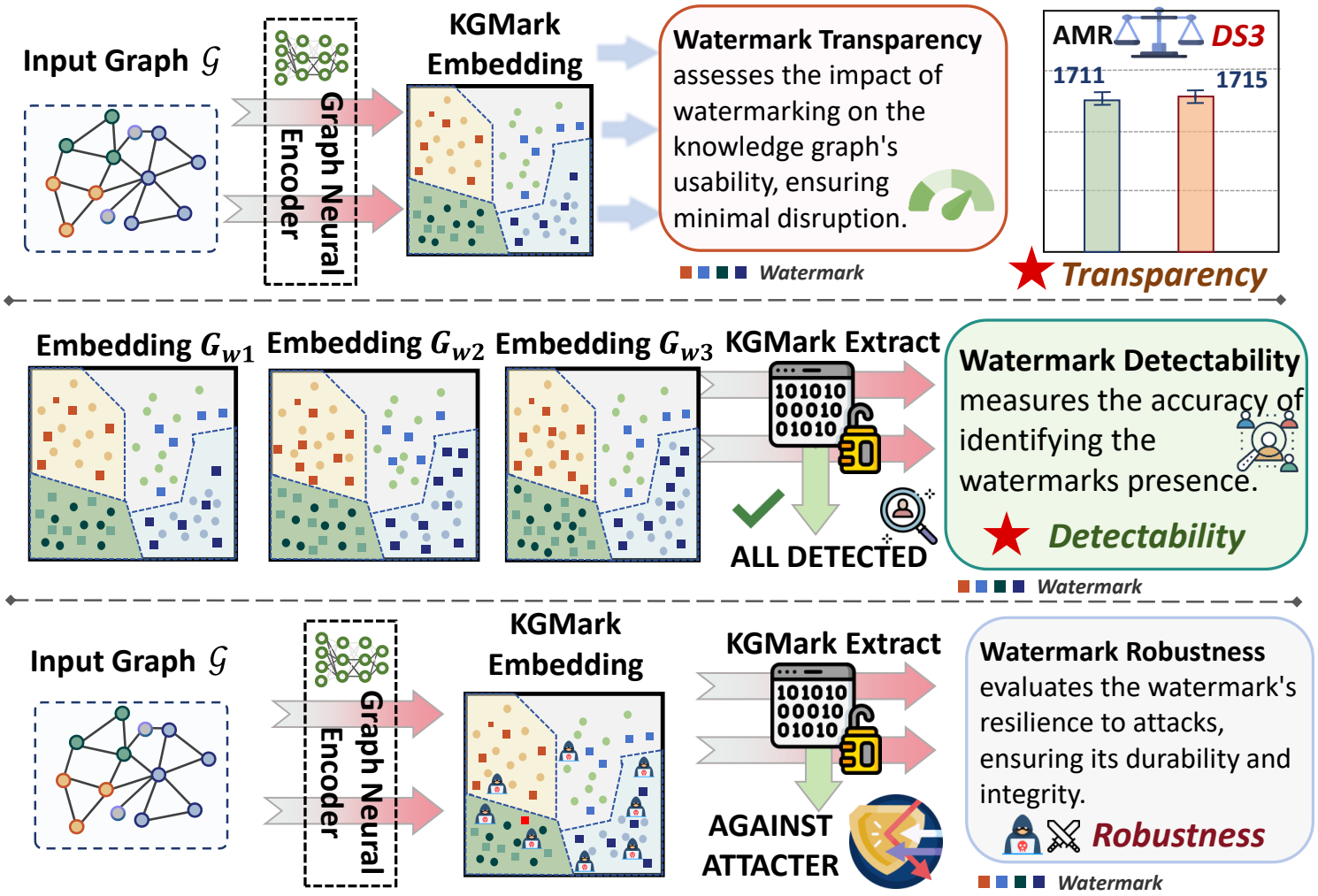

Figure: KGMark ensures Transparency, Detectability, and Robustness.

Methodology: KGMark's Pipeline

The framework embeds and extracts the watermark by manipulating the latent space of a diffusion model.

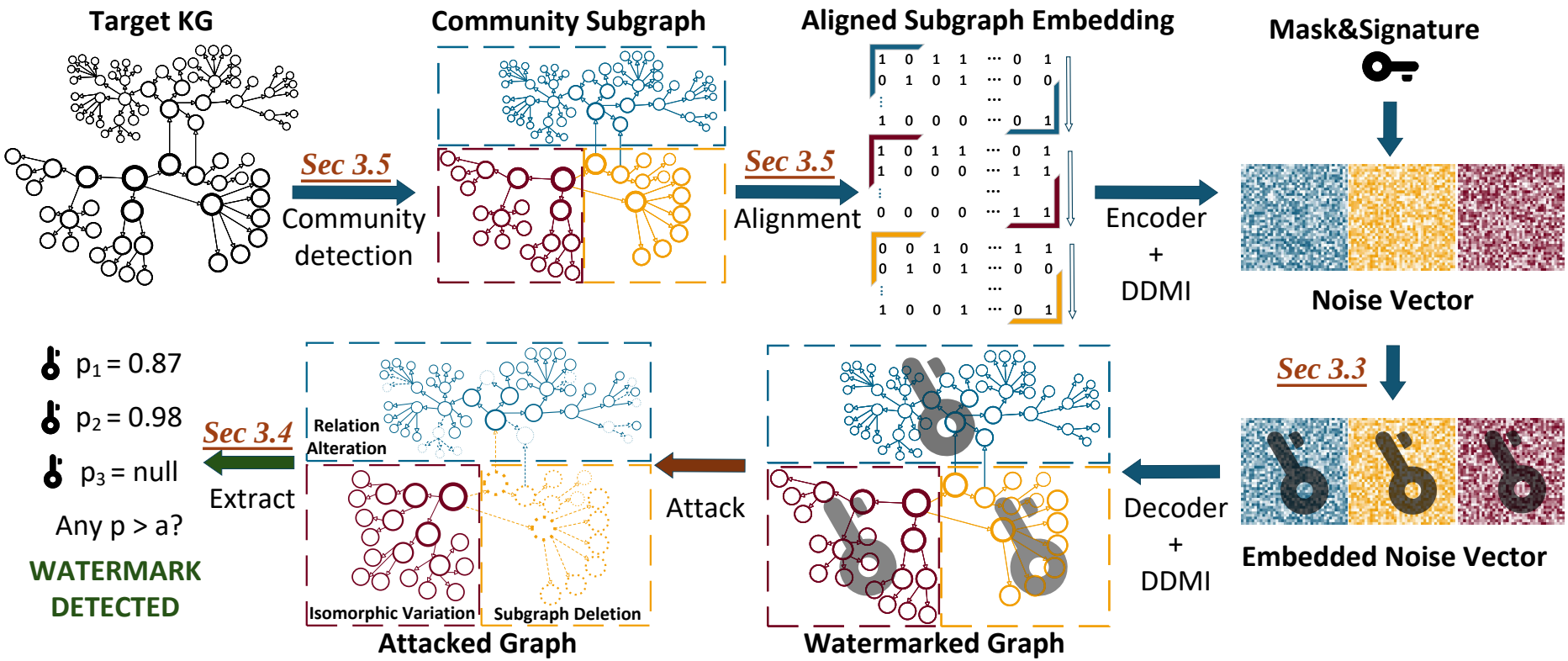

Figure: From a target KG to a watermarked graph, and its verification after potential attacks.

Methodology: KGMark's Pipeline

1. **Pre-processing:** The KG undergoes community detection and graph alignment to normalize its structure.
2. **Embedding:** The graph is encoded into a latent representation Z_0. DDIM is used to get the initial noise vector Z_T.
3. **Watermark Injection:** A signature S is embedded into the frequency domain of Z_T using a learnable mask M.
4. **Decoding:** The watermarked noise vector Z_T^w undergoes reverse diffusion to generate the watermarked graph $\mathcal{G}^w$.
5. **Extraction:** For a given graph, the process is inverted to extract a potential watermark, which is then verified statistically.

Methodology: Watermark Injection

The watermark is embedded in the frequency domain of the latent noise vector (Z_T) to ensure imperceptibility and robustness. This is achieved before the reverse diffusion process begins.

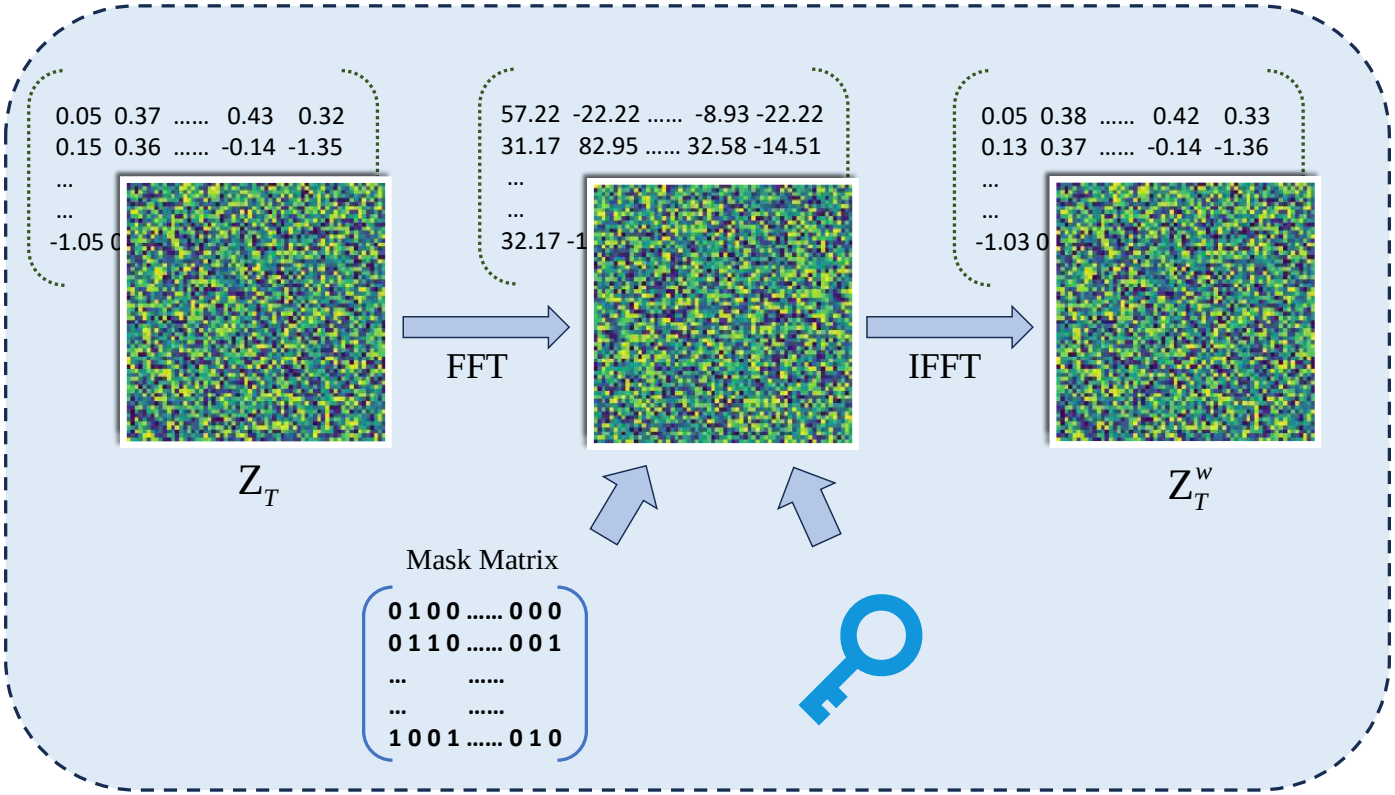

Figure: Visual overview of embedding the watermark in the frequency domain using a mask matrix.

Methodology: Watermark Injection

Injection Process:

1. The Fourier Transform (F) is applied to both the noise vector Z_T and the signature S.
2. The frequency components are combined using the mask M:

$$\Delta = F(Z_T) \cdot (1 - M) + F(S) \cdot M$$

   This replaces selected parts of the noise spectrum with the watermark's spectrum.
3. The Inverse Fourier Transform (F^{-1}) is applied to create the watermarked noise vector $Z_T^w = F^{-1}(\Delta)$.
4. This new vector Z_T^w is then fed into the DDIM's reverse process to generate the final watermarked graph.

Key Innovation I: Learnable Adaptive Watermark Mask (LAWMM)

Goal: Embed the watermark transparently with minimal disruption to the KG's utility.

Challenge:

- Watermark embedding inevitably introduces distortion. How can we control this distortion to preserve the graph's integrity?

Solution: LAWMM

- A **learnable, adaptive mask matrix (LAWMM)** is optimized for each graph's unique structure.
- This mask controls where and how the watermark is injected into the latent space.

Key Innovation I: Learnable Adaptive Watermark Mask (LAWMM)

Mechanism:

- The mask is trained by minimizing the discrepancy between the original and watermarked latent vectors, preserving the *Latent Space Equilibrium*.
- A "sample-then-embed" strategy with a correction term $\alpha\mathcal{S} \cdot \mathbb{M}$ (α is a tunable coefficient) is adopted, ensuring better alignment in the latent space.

$$\mathcal{L} = \sum_{j \in [1,\mathcal{T}]} \left\| \mathcal{Z}^{\text{INV}}_{\mathcal{T}-k_j} - \left[f_w \left(f^{k_j}_{\text{DDIM}}(\mathcal{Z}^{\text{INV}}_{\mathcal{T}}, \mathcal{T}), \mathcal{S}, \mathbb{M} \right) + \alpha\mathcal{S} \cdot \mathbb{M} \right] \right\|^2.$$

- This balances imperceptibility and robustness by placing the watermark in less critical latent regions.

Key Innovation II: Defending Structural Variations

Goal: Ensure the watermark survives attacks unique to graph data.

Isomorphism Variations

- Graphs can have the same structure but different node orderings (isomorphism), which alters their matrix representation.
- **Solution: Graph Alignment.** KGMark normalizes graphs by reordering vertices based on degree and clustering coefficient, creating a canonical representation.

Structural Variations (Attacks)

- Attackers can add/remove edges or nodes to destroy the watermark.
- **Solution: Redundant Hierarchical Embedding.** KGMark partitions the graph into communities and embeds the watermark redundantly across them.
  1. **Global (Community Layer):** Injects into a community's spectral profile.
  2. **Local (Vertex Layer):** Encodes via edge-weights around high-centrality nodes.

Key Innovation III: Likelihood-Based Verification

Goal: Reliably and accurately detect a watermark with statistical confidence.

Challenge:

- Random noise can sometimes resemble a watermark. We need a rigorous method to distinguish a true signal from chance.

Solution: A Hypothesis Test

- Watermark detection is formulated as a statistical hypothesis test:
- $\mathcal{H}_0$ (Null Hypothesis): The graph is clean; extracted noise is standard Gaussian $\mathcal{N}(\mathbf{0}, \sigma^2 \mathcal{I}_{\mathbb{C}})$.
- $\mathcal{H}_1$ (Alternative Hypothesis): The graph is watermarked; noise deviates from the standard.

Key Innovation III: Likelihood-Based Verification

Mechanism:

- To integrate the distance metric $d(\cdot, \cdot)$ and the signature extraction function f_{ex}, we define the residual vector $\mathcal{R}$ as the difference between the extracted signature and the optimal reference signature $\mathcal{K}^* \in \mathcal{S}$.

$$\mathcal{R} = f_{\text{ex}}(\mathbb{M}, \mathcal{G}) - \mathcal{K}^*$$

- A test statistic, $\hat{\mathcal{T}}$, is computed from the extracted signature.

$$\hat{\mathcal{T}} = \frac{1}{\sigma^2} \sum_{i \in \mathbb{M}} |\mathcal{R}_i|^2$$

- Under $\mathcal{H}_0$, this statistic follows a known **noncentral chi-squared (χ^2) distribution**.
- We calculate the p-value: $p = \Pr(\chi^2 \leq \hat{\mathcal{T}} | \mathcal{H}_0)$.
- If p is below a significance level (e.g., 10^{-5}), we confirm the watermark's presence.

Experimental Setup: Datasets & Attacks

Datasets

- Three public KG datasets from diverse domains were used:
- **Alibaba-iFashion (AliF):** E-commerce
- **MIND:** News recommendation
- **Last-FM:** Music

Attacks Evaluated

- Robustness was evaluated against a wide range of post-editing attacks:
- **Structural:** Relation Alteration, Triple Deletion.
- **Noise-based:** Gaussian Noise Injection & Smoothing.
- **Topological:** Graph Isomorphism Variation (IsoVar).
- **Adversarial:** L2 Metric and NEA graph poisoning attacks.

Evaluation Metrics

- **Detectability & Robustness:** Area Under Curve (AUC).
- **Transparency:** Cosine Similarity, GMR, HMR, AMR, and Hits@10.

Experimental Setup: Details & Variants

Implementation Details

- The knowledge graph is first embedded using the **RotatE** model.
- The embedding dimension was set to 4096.
- All experiments were conducted on a single **NVIDIA A800** GPU.

Ablation Variants

- **W/O LAWMM:** Uses a fixed watermark mask matrix instead of the learnable one to test the mask's effectiveness.
- **Only CL:** Applies the watermark exclusively in the Community Layer.
- **Only VL:** Applies the watermark exclusively in the Vertex Layer.

Baselines

- **TreeRing & GaussianShading:** Watermarking methods for diffusion models on images, adapted for graphs.
- **DwtDct & DetQim:** Classical watermarking techniques that modify transformed coefficients.

Experiment Result: Transparency

KGMark preserves the KG's structural integrity and utility for downstream tasks.

Table: Watermark Transparency Result on AliF

Datasets	Method	Cosine Similarity ↑			KG Quality Metric @ 75 Steps			
		50 Steps	65 Steps	75 Steps	GMR ↓	HMR ↓	AMR ↓	Hits@10 ↑
AliF	Original KG	-	-	-	1.828	1.162	135.459	0.8980
	W/O Watermark	0.7971	0.8797	0.9674	3.026	1.579	141.412	0.8318
	DwtDct	0.7215	0.7928	0.8251	5.096	1.699	157.036	0.6933
	DctQim	0.7509	0.7633	0.7653	5.104	1.654	161.142	0.7385
	TreeRing	<u>0.7761</u>	**0.8431**	<u>0.9071</u>	3.928	<u>1.618</u>	152.634	<u>0.8017</u>
	GaussianShading	0.2879	0.3226	0.3538	6.641	1.798	172.813	0.5137
	W/O LAWMM	0.7662	0.7838	0.8643	<u>3.457</u>	1.624	<u>147.305</u>	0.7871
	KGMark	**0.7839**	<u>0.8309</u>	**0.9482**	**3.046**	**1.580**	**141.904**	**0.8296**

Experiment Result: Transparency

Minimal Structural Distortion:

- KGMark achieves high Cosine Similarity scores (e.g., **0.9482** for AliF), indicating the watermarked KG is structurally close to the original.
- This is far better than baselines like GaussianShading (0.3538) and is comparable to a non-watermarked reconstruction (0.9674).

Preserved Downstream Performance:

- Performance on link prediction (Hits@10) is nearly on par with the original KG.
- On AliF, KGMark scores **0.8296** vs. the non-watermarked 0.8318, confirming the KG remains functional.

Experiment Result: Robustness - Structural Attacks

KGMark's hierarchical embedding provides strong resilience against attacks that alter graph structure.

Performance under Attack: (AUC scores on AliF dataset)

Table: Watermark Robustness vs. Structural and Adversarial Attacks

Method	Relation Alteration (50%)	Triple Deletion (50%)	Adversarial	
			L2 Metric	NEA
DwtDct	0.8371	0.7724	0.9577	0.9638
TreeRing	0.7392	0.8091	0.9621	0.9584
Only CL (variant)	0.8864	0.8063	0.9426	0.9535
Only VL (variant)	0.9433	0.8592	0.9521	0.9676
KGMark	**0.9207**	**0.9320**	**0.9841**	**0.9809**

Analysis:

- KGMark maintains a very high AUC (e.g., **0.9320**) even when 50% of graph triples are deleted.
- It significantly outperforms ablated variants, showing that the **combined coarse-grained (community) and fine-grained (vertex) embedding** is essential.

Experiment Result: Robustness - Noise & Smoothing Attacks

The watermark also resists common signal processing attacks applied to graph embeddings.

Performance under Attack: (AUC scores on MIND dataset)

Table: Watermark Robustness vs. Noise and Smoothing

Method	**Gaussian Noise**		**Smoothing**	
	10%	**50%**	**10%**	**50%**
TR	0.86	0.77	0.94	0.83
GaussianShading	0.89	0.85	0.90	0.83
W/O LAWMM	0.98	0.91	0.95	0.89
KGMark	**0.99**	**0.92**	**0.96**	**0.90**

Experiment Result: Robustness - Noise & Smoothing Attacks

Analysis:

- KGMark shows great resilience, with an AUC of **0.92** under 50% Gaussian noise and **0.90** under 50% smoothing.
- The variant without the learnable mask (W/O LAWMM) is also robust, confirming LAWMM is mainly for transparency.
- These results validate KGMark's balance of robustness and usability under aggressive attacks.

Experiment Result: Detectability

KGMark's watermark is highly detectable, but performance depends on key hyperparameters.

DDIM Steps & Significance:

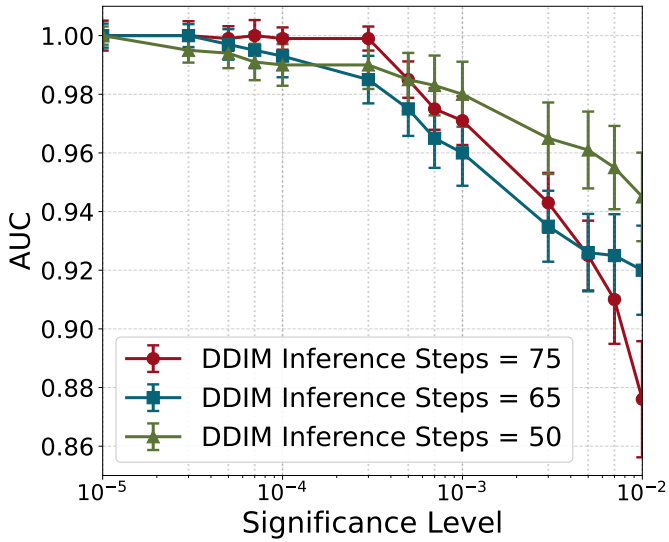

Figure: AUC vs. Sig. Level

- At optimal sig. levels (10^{-5}), more DDIM steps improve AUC.
- At higher sig. levels (10^{-3}), it can increase False Positives.

Experiment Result: Detectability

Stage Alignment:

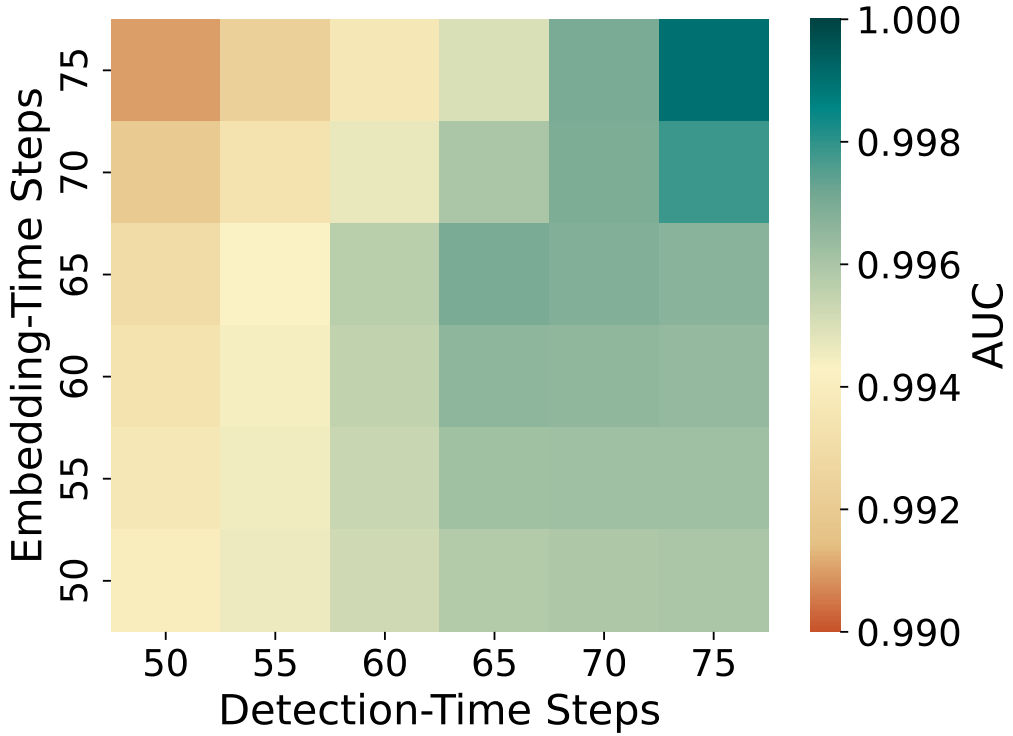

Figure: AUC vs. Emb./Det. Time Steps

- Aligning embedding and detection DDIM steps is **critical**.
- The detection stage has more influence on accuracy.

Case Study: News Recommendation

Question: Does the watermark impact a real-world downstream task?

Experimental Setup:

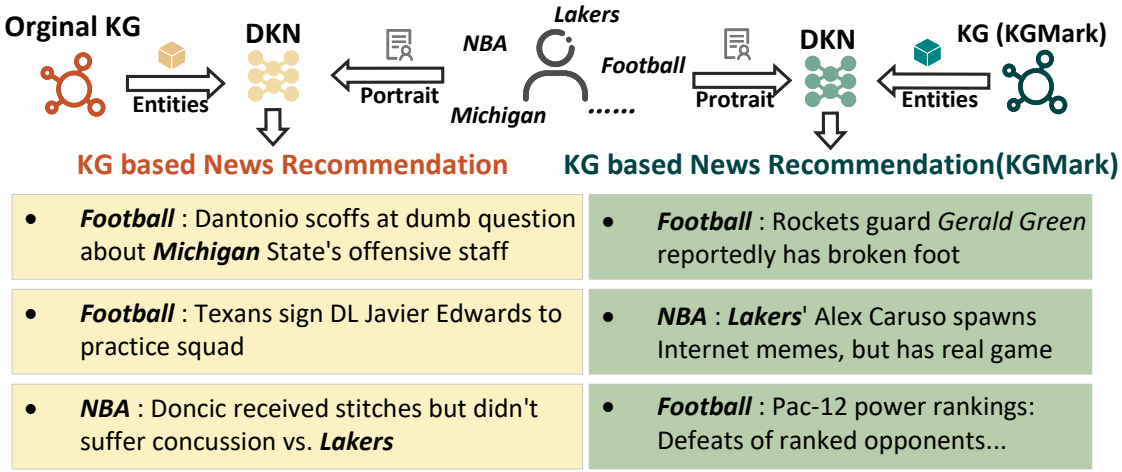

Figure: Comparison of recommendations shows consistent topics and entities.

- **Task:** News recommendation using the DKN model.
- **Dataset:** MIND (large-scale news dataset).
- **Scenario:** Compare recommendations for a sports-focused user using the original vs. watermarked KG.

Case Study: News Recommendation

Findings:

- The content, subcategories, and hot topics of the recommendations remained **consistent**.
- Entity recognition was stable, indicating no degradation in the KG's semantic quality.
- **Conclusion:** The watermark is transparent and does not negatively impact this complex downstream application.

Discussion and Future Work

VAE and Latent Space:

- The choice of graph encoder involves a trade-off: expressive models (like RGAT) can be more vulnerable to certain attacks than simpler, more robust models (like GCN).
- KGMark's design requires balancing representation quality with attack susceptibility.

Future Directions:

- **Embedding Dimensions:** Exploring how embedding size affects the balance between task performance and watermark security.
- **Advanced Samplers:** Investigating compatibility with diffusion samplers beyond DDIM to improve quality and transparency.
- **New Applications:** Extending the framework to other structured data domains like GraphRAG.

Conclusion

Contributions:

- Introduced **KGMark**, the first diffusion watermarking scheme for knowledge graphs.
- Our method successfully provides **robustness**, **transparency**, and **detectability**, protecting synthetic graph data.
- Key innovations include:
  - A **Learnable Adaptive Watermark Mask (LAWMM)** for transparency.
  - A **hierarchical, redundant embedding** strategy for robustness.
  - A **likelihood-based statistical test** for reliable detection.

Impact:

- KGMark provides a foundation for securing the integrity and ownership of synthetic KGs.
- It enables trustworthy use of KGs in applications like recommendation systems and semantic search.

The End

Thanks For Your Listening.