Contents

Integrating Knowledge Graphs with Multi-Modal Machine Learning: A Literature Review

Abstract

Integrating knowledge graphs with multi-modal machine learning (MMML) can align diverse data sources like images, text, audio, and sensors within structured frameworks. However, observed benefits are often fragile: embedding relational graphs may lose meaning, databases can struggle with heavy workloads, and many evaluations focus only on accuracy rather than semantics, latency, or interpretability. This review critically examines integration methods, highlights major trade-offs, and provides practical examples, decision guidelines, and research questions relevant across sectors such as retail, manufacturing, and cultural heritage. It proposes a practical framework—multi-resolution representations, a hybrid graph–vector system, and an objective evaluation template—to promote semantic integrity, realistic operations, and robust governance alongside accuracy.

---

1. Introduction

Multi-modal models align information from text, images, audio, video and tabular signals. Knowledge graphs encode entities, relations, hierarchies and constraints. Intuitively, the two are complementary: KGs provide explicit structure and provenance; MMML provides perceptual generalisation and cross-modal alignment. In practice, integration remains uneven. Many systems append KG embeddings to neural features and claim to be “knowledge-aware”, without demonstrating that directionality, hierarchy or cardinality survive the conversion to vectors. Others introduce joint objectives that better retain semantics, yet falter on scalability or reproducibility. The result is a literature heavy on promise and light on principled engineering.

This review is universal in scope. It is not anchored to a single sector; examples span retail visual search, manufacturing defect retrieval, cultural-heritage image discovery, and education or safety use-cases. The aim is not descriptive breadth for its own sake, but a critical map of *how* KGs help MMML, *where* integration fails and *why*, and *which* design choices reliably produce robust, reproducible improvements. Throughout, we translate claims into worked examples, thresholds and decision rules that can be lifted into any applied project.

2. What Knowledge Graphs Add—and What They Risk

Four concrete contributions

1. **Typed entities**
   Nodes provide named concepts—products, parts, species, artworks, chemicals, events—anchoring multi-modal signals to a controlled vocabulary and preventing synonym sprawl.

2. **Relations and constraints**
   Edges encode asymmetry (*used-for*, *causes*), hierarchy (*is-a*, *part-of*), and composition. These curb the symmetry bias of many contrastive spaces.

3. **Compositional reasoning**
   Paths (e.g., *tool* → *used-for* → *cutting*; *painting* → *created-by* → *artist* → *active-in* → *period*) enable chained filtering and retrieval beyond nearest-neighbour similarity.

4. **Interpretability and auditability**
   Paths and types produce human-readable rationales. Explanations can cite entities and relations, not only heatmaps or token attributions.

Four risks

- **Semantic loss** when discrete relations are compressed into continuous vectors; directionality and one-to-many patterns flatten into proximity.
- **Coverage gaps** where generic KGs lack domain-critical edges; domain drift across organisations and time.
- **Staleness and bias** inherited from sources; multi-modal data amplify skew via nearest-neighbour effects.
- **Operational overhead** in keeping graph facts, embeddings and indices consistent under streaming updates.

A rigorous approach treats **semantic preservation** and **operational realism** as outcomes to measure, not assumptions.

3. Integration Patterns: A Comparative Critique

3.1 KG-as-Features (pre-computed embeddings)

Mechanism. Train a KGE model on the KG; concatenate entity/edge embeddings with text and/or image features; train a lightweight head.

Strengths. Minimal engineering; fast inference; convenient ablations; useful for candidate generation or when attributes align directly with visual/textual cues.

Weaknesses. Weak retention of directionality and hierarchy; embeddings drift as graphs and encoders evolve; brittle out-of-distribution behaviour. Gains can reflect label co-occurrence rather than genuine knowledge use.

Decision. A necessary baseline, valuable for candidate generation, but insufficient for strong claims of knowledge-aware retrieval or reasoning.

3.2 Joint KG–MM Objectives

Mechanism. Learn a shared space for entities and multi-modal features with graph-aware constraints: parents closer than non-ancestors; asymmetric relations remain directional; salient attributes reconstructible from entity vectors.

Strengths. Better semantic fidelity; improved handling of fine-grained hierarchies; more faithful rationales.

Weaknesses. Heavier training; unstable optimisation if losses proliferate; sensitive to coverage gaps; increased complexity for end-to-end reproduction.

Decision. Use when faithful relations matter (explanation, auditing, long-tail classes). Keep constraints minimal and interpretable.

3.3 Retrieval-time Knowledge Routing (hybrid architectures)

Mechanism. At inference, route queries through a **graph store** for structural filtering and rationale paths, and a **vector index** for dense similarity; fuse ranks; cache rationales and hot subgraphs; stream updates to keep components in sync.

Strengths. Specialisation yields scale and latency; instrumentation is straightforward; explanations are first-class (graph paths), not post-hoc.

Weaknesses. More moving parts; eventual consistency across stores; requires clear playbooks for fall-backs and reconciliation.

Decision. The most operationally credible pattern for production retrieval and discovery portals.

4. Universal Application Contexts and Motivating Scenarios

- **Retail visual search.** Query: "espadrille wedge sandal with ankle strap". KG encodes *is-a* and *has-attribute* relations; MM features capture texture and style cues.
- **Manufacturing inspection.** Query: "fatigue crack near rivet on aluminium skin". KG encodes *part-of* and *failure-mode*; images carry visual evidence.
- **Cultural heritage.** Query: "impressionist landscape with cathedral façade". KG encodes *created-by*, *style*, *depicts*; text descriptions complement images.
- **Scientific media.** Query: "fluorescence micrograph of spindle assembly checkpoint proteins". KG encodes *interacts-with*, *located-in*, *process-part-of*.

Across these, long-tail phenomena, hierarchy depth and asymmetric relations make structured knowledge valuable—yet only if preserved and operational at scale.

---

5. Worked Examples (Operational, Reproducible, Domain-agnostic)

Worked Example A — Measuring Semantic Retention

Goal. Determine whether models truly preserve relations essential to meaning.

Protocol.

1. Profile relations: **is-a**, **part-of**, **used-for/causes** (directional), **co-occurs-with** (symmetric); capture arities and depths.
2. Train three variants: baseline MM model; **KG-as-features**; **joint KG–MM** with minimal constraints (hierarchy, asymmetry, attribute reconstruction).
3. Compute **Semantic Retention Score (SRS)** as a composite of four components: **AtP** (Attribute Predictability), **HP** (Hierarchy Presence/precision), **AP** (Asymmetry Preservation), and **RTF** (Relation Type Fidelity). In this review we report SRS as **(AtP, HP, AP, RTF)** and discuss each component's contribution rather than a single undifferentiated score.
4. **Decision rule.** Deploy only if the composite score meets the agreed gate on task-critical relations; otherwise prefer multi-resolution or stronger joint constraints.
   Outcome. Moves "knowledge-aware" from marketing to measurable criterion.

Worked Example B — When to Hybridise Graph and Vector Stores

Goal. Identify when a monolithic property-graph approach fails for embedding-heavy retrieval.
Protocol.

1. Build MMKGs at increasing sizes; attach embeddings alongside graph facts.
2. Benchmark: bulk writes; **two-hop-plus-vector** queries ("filter by relations, then rank by similarity"); mixed read/write under concurrency.
3. Track **p50/p95/p99** latency, memory, index build time and update lag.
   Decision rule. Adopt **hybrid (graph + vector + cache + stream)** once two-hop-plus-vector **p95** exceeds budget at target scale, or memory use breaches safe thresholds; define fall-backs (vector-only short-circuit) and reconciliation windows.
   Outcome. A defensible threshold for architectural choice that generalises across domains.

We evaluate routing and latency at **$N \in \{10^3, 10^4\}$**, reporting **p50/p95/p99** with warmed caches. Larger-scale latency ($\geq 10^5$–10^6) is future work outside this module's scope.

Worked Example C — Domain-specific vs Generic KGs

Goal. Test whether a curated domain KG outperforms a generic KG for open-world retrieval and zero-shot tasks.

Protocol.

1. Fix the MM backbone and training budget.
2. Swap only the KG: **generic** vs **domain-specific** (curated for task-critical relations).
3. Measure retrieval mAP / zero-shot F1 and **rationale-path precision** (proportion of explanations using true paths with correct direction and type).

Decision rule. Prefer the domain KG when accuracy improves by **≥ 3 points** *and* rationale precision by **≥ 10 percentage points**; otherwise enrich the generic KG or narrow scope.

Outcome. Turns "curate a better graph" from hunch to resource-allocation decision.

6. A Practical, Universal Design Framework

6.1 Multi-Resolution Representations

Do not crush all semantics into a single vector. Maintain layered representations and escalate by need:

- **Type layer:** categories and attributes (e.g., product type, material; component category; style; role).
- **Direct-relation layer:** immediate edges with direction and cardinality (**is-a**, **part-of**, **used-for**).
- **Neighbourhood layer:** k-hop summaries for context (co-usage, co-occurrence, workflows).
- **Full-semantics layer:** path extraction for explanation and adjudication.
  Query strategy. Generate candidates with vectors; verify with direct relations or neighbourhood checks; explain with paths on demand. This preserves meaning while meeting latency budgets.

6.2 Hybrid Graph–Vector Architecture

Role split.

- **Graph database** as semantic spine: entities, typed relations, schema, provenance and governance; traversal and path extraction for filtering and explanation.
- **Vector index** for dense similarity across learned embeddings at scale.
- **Cache** for hot subgraphs and recurrent rationales.
- **Stream** for updates and index synchronisation.
- **Operational policy.** Define service-level objectives (e.g., **p95 < 150 ms** for two-hop-plus-vector queries at target scale) and fall-backs (vector-only short-circuit on budget breach; deferred path explanation). Instrument **p50/p95/p99** latency and update lag.
  Practical hierarchy coverage (HP). To raise **HP** without brittle manual curation, generate **conservative is-a edges** from **high-precision regex rules** over **observed** concept names (auto-taxonomy) and union these with a small curated slice. This improves coverage where data exists, reduces spurious parents, and preserves explainability.

6.3 Honest Evaluation (as policy, not an afterthought)

Each experiment should ship with a one-page sheet:

- **Effectiveness:** retrieval/zero-shot metrics on in-domain and shifted distributions, with salient sub-groups.
- **Semantic retention: AtP/HP/AP/RTF** with per-relation breakdowns.
- **Efficiency: p50/p95/p99** latency, memory, index-build times, update lag.
- **Robustness:** modality dropout, label noise, distribution shift, class imbalance.
- **Reproducibility:** seeds, configs, budgets, and spread (mean/standard deviation, min/max).
- **Interpretability:** rationale precision and counterfactual sensitivity (removing the cited path must materially affect the output).

---

7. Common Pitfalls and How to Avoid Them

1. **Interpretability theatre.** Attractive graph overlays that do not affect predictions when removed are decorative, not explanatory. Require faithful rationales whose ablation changes outputs.
2. **Benchmark myopia.** Static leaderboards hide brittleness. Include stress tests on scale, skew, shift and noise; publish degradation curves and tails.
3. **One-store absolutism.** Forcing topology and dense similarity into one database invites slow queries or impoverished semantics. Specialise and route.
4. **Generic-KG complacency.** Breadth without critical relations is a liability. Measure relation criticality; invest where returns are highest.
5. **Over-engineered loss functions.** Many-term joint objectives can be unstable and hard to reproduce. Keep constraints minimal and interpretable (hierarchy, asymmetry, attribute reconstruction) with principled weighting.

8. Ethics, Bias and Governance

KGs and multi-modal datasets encode the assumptions and gaps of their sources. To minimise harm and improve trust:

- **Parity audits** across salient sub-groups (categories, geographies, time).
- **Temporal validity**: time-stamp edges; warn on outdated relations; prefer time-consistent facts for training and explanation.
- **Data minimisation** and privacy-aware retrieval where applicable.
- **Fail-safe behaviours**: abstain or revert to conservative retrieval when knowledge is sparse or contradictory.
- **Provenance and governance**: record sources, curation decisions and versioning; apply change control to graph edits and index rebuilds.

Ethics here is operational: it shapes architecture, evaluation and release gates from day one.

---

9. Research Questions with Measurable Targets

To anchor contributions in evidence, set explicit gates that reflect realistic scope.

- **RQ1 — Knowledge fidelity.**
  Target: Achieve **SRS ≥ 0.75** by the intermediate milestone with **AtP ≥ 0.95**, **HP ≥ 0.25**, **AP ≥ 0.99**, using **AtP/HP/AP/RTF** as reported components.
- **RQ2 — Architecture choice.**
  Target: For **N ∈ {10^3, 10^4}**, instrument **p50/p95/p99** latency for two-hop-plus-vector queries; adopt **hybrid** once **p95** exceeds budget or memory bounds are breached; document fall-backs and reconciliation windows.
- **RQ3 — Task effectiveness.**
  Target: When adding **KG-as-features** to a text baseline, demonstrate **≥ +3 percentage-point micro-F1** (or equivalent retrieval gain) without violating latency budgets.
- **RQ4 — Honest evaluation.**
  Target: Under modest stressors (taxonomy off / unit noise / class shift), contain relative performance drop to **≤ 10%**, and show at least one accuracy-leading baseline is re-ranked when semantics and latency are considered.

10. Summary Table of Critical Findings and Decisions

Approach	***When it works well***	***When it struggles***	***Our solution / decision rule***
KG-as-features	Rapid baselines; simple ablations; cheap inference	Poor directionality/hierarchy retention; drift	Use for candidate generation and as a baseline; deploy only if **SRS ≥ gate**
Joint objectives	Tasks needing faithful relations & fine-grained labels	Unstable optimisation; coverage gaps; heavy training	Minimal, interpretable constraints (hierarchy, asymmetry, attributes); gate by SRS & latency
Monolithic graph DB	< ~10^5 entities; traversal-heavy prototyping	Scale, latency, mixed workloads with embeddings	**Hybrid** (graph for structure; vector for similarity; cache + stream) with **p95 SLOs**
Retrieval-time routing	Production retrieval with explanations	Consistency complexity; ops overhead	Playbooks: vector-only short-circuit on budget breach; scheduled reconciliation; rationale caching

---

11. Limitations and Future Directions

- **Long-range logic and uncertainty.** Multi-resolution designs capture local to mid-range structure; richer reasoning over long causal chains and uncertainty may require symbolic or probabilistic components.
- **Dynamic updates.** Keeping embeddings, indices and graph facts synchronised as entities and relations evolve is a non-trivial engineering challenge.
- **Multilingual and multi-domain semantics.** Aligning cross-language labels and domain-specific jargon with images, audio and text remains open.
- **Edge confidence and scalable explanation.** Many graphs treat edges as binary; real-world relations are graded. Future work should represent and expose confidence in edges and rationales.

12. Synthesis and Recommendations

A system that merely “adds a KG” risks fragile gains and brittle explanations. Integration becomes compelling when reframed as deliberate engineering with three pillars:

1. **Preserve meaning by design.** Use minimal, principled constraints to protect hierarchy and asymmetry; measure **AtP/HP/AP/RTF** and gate deployment on SRS; adopt multi-resolution semantics to avoid flattening.
2. **Scale by specialisation.** Keep a graph store as the semantic spine for structure, governance and paths; offload dense similarity to a vector index; add cache and stream layers; define **SLOs** and fall-backs; instrument **p50/p95/p99**.
3. **Evaluate honestly.** Treat robustness, tails, interpretability and latency as non-negotiable. Publish procedures (splits, seeds, budgets) as well as scores.

Conclusion

The salient question is no longer whether a knowledge graph can be fused with multi-modal models, but whether it can be done while preserving meaning, meeting operational realities, and standing up to scrutiny. The answer is yes—provided we (i) measure semantics explicitly, (ii) architect for scale with specialised components and clear fall-backs, and (iii) evaluate under conditions that resemble real use. With these commitments, open-world retrieval and zero-shot systems can move beyond accuracy-only narratives to deliver transparent, reliable tools—models that not only retrieve and classify, but also **explain** in coherent, human-readable terms.

References

1. Baltrušaitis, T., Ahuja, C. and Morency, L.P. (2019) 'Multimodal machine learning: A survey and taxonomy', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2), pp. 423–443. Available at: https://doi.org/10.1109/TPAMI.2018.2798607

2. Chen, Z. et al. (2024) 'Knowledge Graphs Meet Multi-Modal Learning: A Comprehensive Survey', arXiv preprint, arXiv:2402.05391. Available at: https://arxiv.org/abs/2402.05391

3. Das, A., Rad, P. and Choo, K.K.R. (2020) 'Explainable AI for knowledge graphs: A survey', *ACM Computing Surveys*, 53(4), article 1. Available at: https://doi.org/10.1145/3397512

4. Fey, M. and Lenssen, J.E. (2019) 'Fast graph representation learning with PyTorch Geometric', *arXiv preprint*, arXiv:1903.02428. Available at: https://arxiv.org/abs/1903.02428

5. García-Durán, A. and Niepert, M. (2018) 'Learning graph representations with embedding propagation', in *Advances in Neural Information Processing Systems 31*, pp. 5119–5130. Available at: https://arxiv.org/abs/1710.03059

6. Ji, S. *et al.* (2022) 'A survey on knowledge graphs: Representation, acquisition, and applications', *IEEE Transactions on Neural Networks and Learning Systems*, 33(2), pp. 494–514. Available at: https://doi.org/10.1109/TNNLS.2020.2978386

7. Krishna, R. *et al.* (2017) 'Visual Genome: Connecting language and vision using crowdsourced dense image annotations', *International Journal of Computer Vision*, 123(1), pp. 32–73. Available at: https://doi.org/10.1007/s11263-016-0981-7

8. Liu, X. *et al.* (2021) 'Multi-task deep neural networks for natural language understanding', in *Proceedings of ACL 2021*, pp. 4487–4496. Available at: https://aclanthology.org/2021.acl-long.349

9. Marino, K., Salakhutdinov, R. and Gupta, A. (2017) 'The more you know: Using knowledge graphs for image classification', in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 20–28. Available at: https://arxiv.org/abs/1612.04844v2

10. Mehrabi, N. *et al.* (2021) 'A survey on bias and fairness in machine learning', *ACM Computing Surveys*, 54(6), article 115. Available at: https://doi.org/10.1145/3457607

11. Nickel, M. *et al.* (2016) 'A review of relational machine learning for knowledge graphs', *Proceedings of the IEEE*, 104(1), pp. 11–33. Available at: https://doi.org/10.1109/JPROC.2015.2483592

12. Paulheim, H. (2017) 'Knowledge graph refinement: A survey of approaches and evaluation methods', *Semantic Web*, 8(3), pp. 489–508. Available at: https://doi.org/10.3233/SW-160218

13. Radford, A. *et al.* (2021) 'Learning transferable visual models from natural language supervision', in *Proceedings of ICML 2021*, pp. 8748–8763. Available at: https://proceedings.mlr.press/v139/radford21a.html

14. Shen, Y. *et al.* (2021) 'Medical knowledge graphs for precision medicine: A survey', *Briefings in Bioinformatics*, 22(4), pp. 1–15. Available at: https://doi.org/10.1093/bib/bbaa120

15. Trivedi, R. *et al.* (2017) 'Know-Evolve: Deep temporal reasoning for dynamic knowledge graphs', in *Proceedings of ICML 2017*, pp. 3462–3471. Available at: https://proceedings.mlr.press/v70/trivedi17a.html

16. Veličković, P. *et al.* (2018) 'Graph attention networks', *arXiv preprint*, arXiv:1710.10903. Available at: https://arxiv.org/abs/1710.10903

17. Wang, Q. *et al.* (2017) 'Knowledge graph embedding: A survey of approaches and applications', *IEEE Transactions on Knowledge and Data Engineering*, 29(12), pp. 2724–2743. Available at: https://doi.org/10.1109/TKDE.2017.2754499

18. Wang, P. *et al.* (2020) 'Knowledge-enabled visual question answering', in *Proceedings of AAAI 2020*, pp. 12223–12230. Available at: https://ojs.aaai.org/index.php/AAAI/article/view/6859

19. Yao, L., Mao, C. and Luo, Y. (2019) 'KG-BERT: BERT for knowledge graph completion', *arXiv preprint*, arXiv:1909.03193. Available at: https://arxiv.org/abs/1909.03193

20. Zhang, W. *et al.* (2019) 'Multimodal knowledge graph embeddings for cross-modal retrieval', in *Proceedings of SIGIR 2019*, pp. 385–394. Available at: https://doi.org/10.1145/3331184.3331194

21. Zhou, H. *et al.* (2020) 'Commonsense knowledge-aware conversation generation with graph attention', in *Proceedings of IJCAI-20*, pp. 4623–4629. Available at: https://doi.org/10.24963/ijcai.2020/642

22. Zhu, Y. *et al.* (2020) 'Deep graph contrastive representation learning', *arXiv preprint*, arXiv:2006.04131. Available at: https://arxiv.org/abs/2006.04131

Industry and Technical Sources

23. Neo4j (2025) *Graph Data Science Library Documentation*. Available at: https://neo4j.com/docs/graph-data-science/current/

24. Neo4j (2025) *Cypher Query Language Manual*. Available at: https://neo4j.com/docs/cypher-manual/current/

Biomedical and Domain-Specific Sources

25. Zhang, Y. *et al.* (2024) 'A comprehensive large scale biomedical knowledge graph for AI powered data driven biomedical research', PMC, 2025. Available at: https://pmc.ncbi.nlm.nih.gov/articles/PMC10760044/

26. Wang, L. *et al.* (2023) 'Petagraph: A large-scale unifying knowledge graph framework for integrating biomolecular and biomedical data', *Scientific Data, 2024*. Available at: https://www.nature.com/articles/s41597-024-04070-w