IsnadGuard: Detecting Fabricated Chains of Narration in Hadith Transmission Networks

Ghaleb Aldoboni [*1] **Mohammed Talha Alam** [*1] **Lobna Nassar** [2] **Fakhri Karray** [1]

Abstract

The isnad is the ordered chain of narrators through whom a hadith is transmitted. Hadith science evaluates reports through narrator reliability and chain continuity, yet computational work has largely treated authenticity as text classification over the report content (matn). We study a focused task: detecting and localizing fabricated chains from the narrator-transmission structure alone. Using Sanadset 650K, we construct a directed narrator graph and generate corruptions grounded in classical defect typologies: narrator substitution (*tas-heef fi al-isnad*) and chain splicing (*idraj al-sanad*). We propose ISNADGUARD, a multiple-instance graph model that scores each transition using local statistics and narrator embeddings, aggregates transition scores with noisy-OR, and is trained jointly for chain classification, contrastive ranking, and edge localization. On 43,539 held-out chains, ISNADGUARD improves AUROC from 0.727 to 0.836 and Hit@2 from 0.838 to 0.916 over an edge-frequency baseline.

1. Introduction

For more than thirteen centuries, Islamic scholarship developed a meticulous source-criticism tradition around the isnad, the ordered chain of transmitters through whom a hadith was passed down. Scholars treated provenance as primary evidence: the *rijal* literature catalogued thousands of narrators with biographical and reliability records, while hadith-critical works enumerated recurring chain defects and fabrications such as narrator substitution (*tas-heef fi al-isnad*), inserted sanad segments (*idraj al-sanad*), and chain inversions (*qalb*) (Markaz al-Minhaj lil-Ishraf wa-al-Tadrib al-Tarbawi, 2021; Kamali, 2005). Computationally, this becomes anomaly detection over directed transmission paths: which link in the chain breaks structural plausibility? In graph terms, an isnad is a path in a directed narrator network, a fabricated chain contains an unsupported transition, and the scholar's question reduces to detecting and localizing that broken edge.

Research Gap. Computational work on hadith has mostly bypassed this question. Existing models classify reports from the matn or topic metadata, conflating the chain with its content and offering no edge-level account of what makes a transmission suspicious. Authenticity classifiers predict document-level labels from text or mixed features (Haque et al., 2020; Najeeb, 2020; Tarmom et al., 2022; Refaee, 2022; Gaanoun & Alsuhaibani, 2022; Alghamdi et al., 2025), while generic graph anomaly detection targets node, edge, or subgraph anomalies rather than ordered paths (Kim et al., 2022; Tang et al., 2022). Neither line treats the isnad as what it is, a path in a transmission graph whose label can be decided by a single unsupported transition.

Related Work. Prior computational work has built corpora and knowledge graphs around this tradition, most notably Sanadset 650K (Mghari et al., 2022), AR-Sanad 280K (Mahmoud et al., 2022), Multi-IsnadSet (Farooqi et al., 2024), and knowledge-graph resources (Kamran et al., 2023; Mahmoud et al., 2024), along with narrator-representation methods (Mghari et al., 2024). Recent Islamic-domain ML has also moved toward culturally grounded evaluation and multimodal religious datasets (Sayeed et al., 2025; Salman et al., 2025). Standard graph learners such as GAT and GraphSAGE (Veličković et al., 2018; Hamilton et al., 2017) target node or subgraph anomalies rather than ordered paths, and multiple-instance learning (Dietterich et al., 1997) supplies the formal frame for the at-least-one-fault semantics that distinguish fabricated chains from valid ones.

Contribution. We introduce ISNADGUARD, the first matn-free benchmark and model for fabricated-isnad detection on a narrator-transmission graph. We formalize the task as path-level anomaly detection with edge-level localization, and release an evaluation suite of 43,539 chains from Sanadset 650K paired with five corruption operators grounded in classical defect typologies. ISNADGUARD scores each tran-

[*]Equal contribution [1]Department of Machine Learning, Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, UAE [2]College of Engineering, American University of Ras Al Khaimah (AURAK), Ras Al Khaimah, UAE. Correspondence to: Ghaleb Aldoboni <ghaleb.aldoboni@mbzuai.ac.ae>.

Proceedings of the 43^{rd} International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026.

sition with local graph statistics and narrator embeddings, aggregates the scores under at-least-one-fault semantics, and is trained jointly for chain classification, contrastive ranking, and edge localization. On the full test set, ISNADGUARD raises AUROC from 0.727 to 0.836 and Hit@2 localization from 0.838 to 0.916 over an edge-frequency baseline, with the largest gains on substitution-style corruptions and consistent improvement on the harder splice setting.

2. Method

2.1. Problem Formulation

An *isnad chain* is an ordered sequence $c = (n_1, \ldots, n_L)$ with $n_i \in \mathcal{N}$ and $L \geq 2$, inducing $L-1$ directed transitions $e_i = (n_i, n_{i+1})$. Chain labels are $y(c) = 1$ for synthetically corrupted chains and $y(c) = 0$ for attested ones; for corrupted chains the corruption process additionally returns $\mathcal{B}(c) \subseteq \{1, \ldots, L-1\}$, the index set of injected transitions, used as edge-level supervision. The model outputs a chain-level anomaly score $P_{\text{bad}}(c) \in [0, 1]$ and an edge ranking summarized by $i^\star(c) = \arg\max_i q_i(c)$.

2.2. Narrator-Transmission Graph

From the training split we build a directed weighted graph $G_{\text{train}} = (V, E, w)$ in which $V \subseteq \mathcal{N}$ are narrators, $(u, v) \in E$ whenever some training chain contains u immediately followed by v, and $w(u, v)$ counts such occurrences. All graph statistics are computed strictly from G_{train} to prevent leakage; narrators seen only at evaluation time map to a shared out-of-vocabulary embedding.

2.3. Synthetic Corruptions

For each attested chain c^{att} we generate a corrupted counterpart $\tilde{c}$ from one of five operators. Four are substitution-style, instantiating *tas-heef fi al-isnad*: **(C1) random** replacement with a uniformly sampled narrator; **(C2) position-matched** replacement controlling for positional priors; **(C3) degree-matched** replacement controlling for popularity; and **(C4) hard-neighbor** replacement from the k-hop graph neighborhood, producing globally plausible but locally unsupported transitions. The fifth, **(C5) chain splice**, instantiates *idraj al-sanad* by joining an attested prefix to an attested suffix at a shared join narrator drawn from disjoint contexts. Each operator records $\mathcal{B}(\tilde{c})$ for supervised localization.

2.4. ISNADGUARD

Per-edge scorer. For transition $e_i = (u, v)$ at position i, we form a feature vector $\boldsymbol{\phi}_i = \boldsymbol{\phi}(u, v, c, i; G_{\text{train}})$ summarizing edge counts, node degrees, conditional forward and backward transition frequencies, normalized position, boundary indicators, and left and right context counts. Combined with learnable narrator embeddings $\mathbf{h}_u, \mathbf{h}_v \in \mathbb{R}^{d_h}$, the full edge representation is

$$\mathbf{z}_i = [\boldsymbol{\phi}_i;\ \mathbf{h}_u;\ \mathbf{h}_v;\ |\mathbf{h}_u - \mathbf{h}_v|;\ \mathbf{h}_u \odot \mathbf{h}_v],$$

and an MLP f_θ produces the edge anomaly logit $a_i = f_\theta(\mathbf{z}_i)$ with probability $q_i = \sigma(a_i)$.

Algorithm 1 ISNADGUARD: training and inference

Input: attested chains $\{c_k^{\text{att}}\}$, training graph G_{train}
Build graph/path features and initialize narrator embeddings and MLP parameters
for each minibatch of attested chains **do**
  Sample corruption types and produce paired corrupted chains $\tilde{c}$
  for each chain $c \in \{c^{\text{att}}, \tilde{c}\}$ and each transition e_i **do**
    Compute $\mathbf{z}_i$, $a_i = f_\theta(\mathbf{z}_i)$, and $q_i = \sigma(a_i)$
  end for
  Compute $P_{\text{bad}}(c) = 1 - \prod_i (1 - q_i)$
  Compute $\mathcal{L}_{\text{cls}}$, $\mathcal{L}_{\text{rank}}$, and $\mathcal{L}_{\text{loc}}$
  Update model parameters
end for
Inference: return $P_{\text{bad}}(c)$ and $i^\star = \arg\max_i q_i$

Chain aggregation. A corrupted chain may be caused by a single bad transition, so we aggregate per-edge probabilities by noisy-OR:

$$P_{\text{bad}}(c) = 1 - \prod_{i=1}^{L-1} (1 - q_i).$$

2.5. Training Objective

For each attested chain c^{att} we sample a corrupted partner $\tilde{c}$ and optimize $\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda_r \mathcal{L}_{\text{rank}} + \lambda_\ell \mathcal{L}_{\text{loc}}$. $\mathcal{L}_{\text{cls}}$ is the standard binary cross-entropy on $y(c)$ against $P_{\text{bad}}(c)$. The ranking term enforces a margin between paired chains,

$$\mathcal{L}_{\text{rank}} = \max\big(0,\ m + P_{\text{bad}}(c^{\text{att}}) - P_{\text{bad}}(\tilde{c})\big)\,,$$

and the localization term concentrates anomaly mass on injected edges via softmax cross-entropy

$$\mathcal{L}_{\text{loc}} = -\frac{1}{|\mathcal{B}(\tilde{c})|} \sum_{i \in \mathcal{B}(\tilde{c})} \log \pi_i, \quad \pi_i = \frac{\exp(a_i)}{\sum_j \exp(a_j)}.$$

The three terms are complementary: classification shapes the chain probability, ranking widens the attested–corrupted gap, and localization aligns the maximum edge score with the structurally guilty transition. Algorithm 1 summarizes training and inference.

3. Experiments

3.1. Data and Setup

We process Sanadset 650K by discarding chains with missing `SANAD` fields, "No SANAD" placeholders, and chains

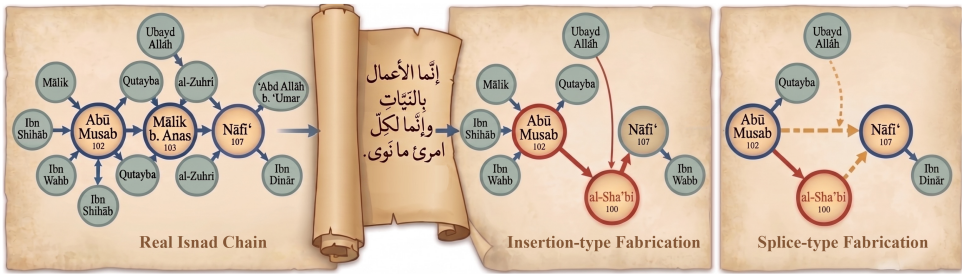

Figure 1. Fabricated-chain detection on a Sanadset narrator-graph neighborhood. A fabricated chain can reuse real narrator nodes and plausible subpaths while introducing a local transition with no network support. The object of interest is the claimed edge, not the narrator identity; this edge-level view motivates our multiple-instance path anomaly model.

shorter than two narrators, and we lightly normalize Arabic narrator names by merging diacritic and orthographic variants. The resulting corpus contains 435,376 chains over 183,268 unique narrators, split 80/10/10 into 348,300 training, 43,537 development, and 43,539 test chains. The training graph has 686,793 directed edges. Each development and test chain is paired with one corrupted partner from each of the five operators in Sec. 2.3, giving 217,685 evaluation pairs per split.

3.2. Evaluation Protocol and Baselines

We report thresholded accuracy at the development-tuned operating point, AUROC as the primary chain-level ranking metric, and Hit@k for edge-level localization, measuring whether at least one injected transition lies among the top-k most-anomalous edges of the corrupted chain. Because the task is ranking and localization rather than fixed-threshold classification, AUROC and Hit@k carry the primary weight.

We compare against four baselines spanning the natural design space. *Edge frequency* scores each transition by $\log(1 + w(u, v))$ and aggregates by chain mean; it is uncalibrated but captures the dominant statistical signal. *Graph-feature MLP* uses the same ϕ_i features as ISNADGUARD but no narrator embeddings, isolating the contribution of identity (Henderson et al., 2012). ISNADGAT and ISNADSAGE are compact full-batch implementations of GAT (Veličković et al., 2018) and GraphSAGE (Hamilton et al., 2017) trained with positive-class weighting, probing whether neighborhood aggregation suffices for this path-local task.

3.3. Implementation Details

Narrator embeddings are 64-dimensional; the per-edge MLP has two hidden layers of width 128 with ReLU activations. We optimize with AdamW at learning rate 10^{-3}, weight decay 10^{-5}, batch size 256 chain pairs, and gradient clipping at norm 1. Loss weights are $\lambda_r = 0.5$ and $\lambda_\ell = 1.0$ with margin $m = 0.2$, selected by a coarse development-set sweep. Training runs for 20 epochs on a single NVIDIA RTX 3090 in approximately four hours. All test numbers are reported after fixing the operating point and all hyperparameters on the development set.

Table 1. Sanadset test performance. All metrics are higher-is-better. Best values are bolded; second-best values are underlined.

Model	Acc.	AUROC	Hit@1	Hit@2
Baselines				
Edge frequency	**0.909**	0.727	0.676	0.838
Graph feat. MLP	**0.909**	0.748	<u>0.741</u>	<u>0.873</u>
ISNADGAT	<u>0.751</u>	0.600	0.486	0.664
Ours				
ISNADGUARD	**0.909**	0.786	**0.790**	**0.916**
ISNADGUARD+FREQ.	**0.909**	<u>0.801</u>	**0.790**	**0.916**
ISNADGUARD-CALIB.	**0.909**	**0.836**	**0.790**	**0.916**

3.4. Main Results

Table 1 reports test performance for all models. Edge frequency is a deceptively strong baseline after thresholding (0.909 accuracy), reflecting that the marginal distribution of transitions is itself a useful authenticity prior, yet it ranks chains weakly (0.727 AUROC) and localizes inconsistently (0.676 Hit@1). The graph-feature MLP closes part of the gap (0.748 AUROC) by combining features the frequency baseline ignores. ISNADGUARD, adding narrator embeddings and the joint contrastive–localization objective, raises AUROC to 0.786 and Hit@2 to 0.916, a 7.8-point absolute improvement on localization. The convex blend with edge frequency lifts AUROC to 0.801, and chain-level calibration on top of the trained model reaches 0.836 AUROC while preserving the edge ranking and therefore Hit@k. ISNADGAT underperforms all alternatives at 0.600 AUROC, consistent with global neighborhood aggregation washing out the local path-specific signal this task depends on.

3.5. Ablations

Table 2 disentangles ISNADGUARD's components. Removing narrator embeddings or path-context features each costs roughly 0.7 AUROC points and 1.5–3 Hit@1 points, indicating that the two information sources are largely comple-

Table 2. Ablations on the full Sanadset test set. All metrics are higher-is-better. Best values are bolded and lightly shaded; second-best values are underlined.

Variant	AUROC	Hit@1	Hit@2
ISNADGUARD	0.786	**0.790**	**0.916**
w/o narrator emb.	0.779	<u>0.773</u>	<u>0.909</u>
w/o path context	0.779	0.762	0.906
w/o localization loss	0.749	0.760	0.902
Mean aggregation	**0.811**	**0.790**	**0.916**
Max aggregation	<u>0.801</u>	**0.790**	**0.916**

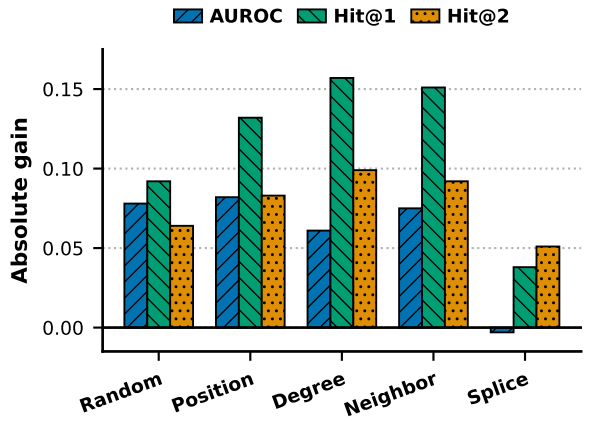

Figure 2. Absolute gain over the edge-frequency baseline by corruption type. Gains are largest for substitution-style corruptions (C1–C4), while splice (C5) is hardest.

mentary. The single most damaging change is removing the localization loss: AUROC drops to 0.749 and Hit@1 to 0.760. The mechanism is the noisy-OR aggregator: because $P_{\text{bad}}(c)$ is dominated by $\max_i q_i$, sharpening the maximum directly improves chain-level discrimination, and the localization loss does exactly that. We also compare two post-training aggregators (mean and max) to the noisy-OR default: both improve AUROC (to 0.811 and 0.801) without changing the edge ranking. We retain noisy-OR as the primary aggregator for its probabilistic interpretation.

3.6. Per-Corruption Analysis

Figure 2 breaks down test performance by corruption family. ISNADGUARD improves Hit@2 across all five, with the largest gains on degree-matched ($+0.099$) and hard-neighbor ($+0.093$) substitutions, which are precisely the corruptions designed to defeat trivial graph features. The pattern is informative: where edge frequency suffers most, learned embeddings help most, consistent with embeddings capturing narrator-pair compatibility that raw frequency cannot. The splice setting (C5) is the hardest by a clear margin (0.724 AUROC, statistically tied with edge frequency), because by construction both halves of a spliced chain are attested subpaths and only the bridge edge is locally synthetic. Even there, Hit@2 improves by $+0.052$, showing that a single locally anomalous edge in an otherwise valid path remains recoverable.

4. Discussion

Path-local versus neighborhood-aggregation models. The ISNADGAT result is the cleanest signal in the experiments: standard graph neural networks, designed to aggregate information over neighborhoods and produce node representations, are a poor fit for a task that asks whether a specific directed edge fits a specific ordered path. ISNADGUARD uses the same underlying graph information but consults it in a path-local way through per-edge features and pairwise embedding interactions, which appears to be the right inductive bias for fabricated-isnad detection.

Why localization supervision helps ranking. A natural reading of the ablation is that supervised localization is the load-bearing component of the loss. Although it operates only on corrupted chains and only at known broken indices, removing it costs more AUROC than removing any feature group. The cause is the aggregator: $P_{\text{bad}}(c)$ is monotone and max-dominated under noisy-OR, so any procedure that concentrates anomaly mass on a single edge improves chain-level discrimination. Localization supervision is the most direct way to do this, and chain classification alone does not exert the same pressure.

Limitations. The benchmark is synthetic, and the splice setting marks the limit of structural-only detection: when both halves of a chain are real, only the bridge edge betrays the fabrication, so a curated evaluation set built with hadith specialists is a natural next step. Narrator disambiguation is a further confound, as shared names may collapse to a single node despite distinct historical persons; rijal-grounded identity resolution and temporal–biographical features (birth/death dates, geographic proximity, teacher–student admissibility windows) would let the model rule out edges that are graph-plausible but historically impossible.

5. Conclusion

We framed fabricated-chain detection in hadith transmission as a path-level anomaly task and introduced ISNADGUARD, a multiple-instance graph model for chain classification and edge localization. On 43,539 chains from Sanadset 650K, it improves AUROC from 0.727 to 0.836 and Hit@2 from 0.838 to 0.916 over an edge-frequency baseline, while the localized edge scores make decisions auditable by identifying which narrator transition breaks a chain. It complements text-based authenticity research and offers a basis for integrating scholar annotations with biographical rijal evidence.

References

Alghamdi, J., Albukhari, A., and Al-Dala'in, T. Pretrained models against traditional machine learning for detecting fake hadith. *Electronics*, 14(17):3484, 2025. doi: 10.3390/electronics14173484.

Dietterich, T. G., Lathrop, R. H., and Lozano-Pérez, T. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31–71, 1997. doi: 10.1016/S0004-3702(96)00034-3.

Farooqi, A. M., Malick, R. A. S., Shaikh, M. S., and Akhunzada, A. Multi-isnadset mis for sahih muslim hadith with chain of narrators, based on multiple isnad. *Data in Brief*, 54:110439, 2024. doi: 10.1016/j.dib.2024.110439.

Gaanoun, K. and Alsuhaibani, M. Fabricated hadith detection: A novel matn-based approach with transformer language models. *IEEE Access*, 10:113330–113342, 2022. doi: 10.1109/ACCESS.2022.3217457.

Hamilton, W. L., Ying, R., and Leskovec, J. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Haque, F., Orthy, A. H., and Siddique, S. Hadith authenticity prediction using sentiment analysis and machine learning. In *2020 IEEE 14th International Conference on Application of Information and Communication Technologies (AICT)*, 2020. doi: 10.1109/AICT50176.2020.9368569.

Henderson, K., Gallagher, B., Eliassi-Rad, T., Tong, H., Basu, S., Akoglu, L., Koutra, D., Faloutsos, C., and Li, L. Rolx: Structural role extraction & mining in large graphs. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1231–1239, 2012. doi: 10.1145/2339530.2339723.

Kamali, M. H. *A Textbook of Hadith Studies: Authenticity, Compilation, Classification and Criticism of Hadith*. The Islamic Foundation, Leicestershire, UK, 2005. ISBN 9780860374350.

Kamran, A. B., Abro, B., and Basharat, A. Semantichadith: An ontology-driven knowledge graph for the hadith corpus. *Journal of Web Semantics*, 78:100797, 2023. doi: 10.1016/j.websem.2023.100797.

Kim, H., Lee, B. S., Shin, W.-Y., and Lim, S. Graph anomaly detection with graph neural networks: Current status and challenges. *IEEE Access*, 10:111820–111829, 2022. doi: 10.1109/ACCESS.2022.3211306.

Mahmoud, S., Saif, O., Nabil, E., Abdeen, M., ElNainay, M., and Torki, M. AR-sanad 280k: A novel 280k artificial sanads dataset for hadith narrator disambiguation. *Information*, 13(2):55, 2022. doi: 10.3390/info13020055.

Mahmoud, S., Nabil, E., Saif, O., and Torki, M. Narrator identification by querying sanad graph and utilizing the narratorskg on AR-sanad 280k-v2 dataset. *Neural Computing and Applications*, 36(36):23169–23180, 2024. doi: 10.1007/s00521-024-10194-2.

Markaz al-Minhaj lil-Ishraf wa-al-Tadrib al-Tarbawi. *Al-Khulasa fi Mustalah al-Hadith*. Markaz al-Minhaj lil-Ishraf wa-al-Tadrib al-Tarbawi, 2021. Arabic title: al-Khulasa fi Mustalah al-Hadith.

Mghari, M., Bouras, O., and El Hibaoui, A. Sanadset 650k: Data on hadith narrators. *Data in Brief*, 44:108540, 2022. doi: 10.1016/j.dib.2022.108540.

Mghari, M., Bouras, O., and El Hibaoui, A. Narrator2vec: An efficient narrator representation in hadith literature using word embedding. *Arabian Journal for Science and Engineering*, 49:4479–4494, 2024. doi: 10.1007/s13369-023-08224-7.

Najeeb, M. M. A. A novel hadith processing approach based on genetic algorithms. *IEEE Access*, 8:20233–20244, 2020. doi: 10.1109/ACCESS.2020.2968417.

Refaee, E. A. Detecting hadith authenticity using a deep-learning approach. *Scientific Journal of King Faisal University: Basic and Applied Sciences*, 23(1):80–84, 2022. doi: 10.37575/b/sci/210084.

Salman, M. U., Qazi, M. A., and Alam, M. T. Quran-md: A fine-grained multimodal dataset of the quran. In *5th Muslims in ML Workshop co-located with NeurIPS 2025*, 2025.

Sayeed, M. A., Alam, M. T., Imam, R., Sohail, S. S., and Hussain, A. From rag to agentic: Validating islamic-medicine responses with llm agents. *arXiv preprint arXiv:2506.15911*, 2025.

Tang, J., Li, J., Gao, Z., and Li, J. Rethinking graph neural networks for anomaly detection. In *International conference on machine learning*, pp. 21076–21089. PMLR, 2022.

Tarmom, T., Atwell, E., and Alsalka, M. Deep learning vs compression-based vs traditional machine learning classifiers to detect hadith authenticity. In *Information Management and Big Data*, volume 1577 of *Communications in Computer and Information Science*, pp. 206–222. Springer, 2022. doi: 10.1007/978-3-031-04447-2_14.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. Graph attention networks. In *International Conference on Learning Representations*, 2018.