EJSIT
European Journal of Science, Innovation and Technology
ISSN: 2786-4936
www.ejsit-journal.com
Volume 6 | Number 4 | 2026

Graph Neural Networks for Fraudulent Transaction Pattern Detection in Digital Lending Systems

Kaleshwar Aryasomayajula
Sr Specialist Software Developer, Charles Schwab,
San Tan Valley, Arizona, USA

ABSTRACT

Digital lending systems handle borrower applications, repayment events, device traces, and payment flows in environments where fraud arises from relationships among actors. A single application record rarely exposes coordinated fraud, while shared devices, repeated accounts, recycled contacts, and post-disbursement transfers often reveal hidden links. This article examines the use of Graph Neural Networks for detecting fraudulent transaction patterns in digital lending, with particular attention to underbanked and thin-file borrowers. The study aims to build an analytical model that connects graph construction, fraud pattern recognition, and decision governance. The material base covers recent studies on financial fraud detection, consumer loan fraud, graph anomaly detection, imbalanced learning and online credit loan risk. Comparative source analysis, conceptual synthesis, and typologization guide the research design. The article identifies graph representation principles, GNN mechanisms for coordinated fraud detection, and implementation rules for credit decision workflows. The proposed model supports auditable fraud controls without replacing creditworthiness assessment.

Keywords: graph neural networks, digital lending, fraud detection, transaction patterns, credit risk, graph anomaly detection, online lending, imbalanced data, underbanked borrowers, decision governance

INTRODUCTION

Digital lending platforms process credit decisions through short application windows, fragmented borrower histories, and high-volume transaction streams. Underbanked borrowers often lack stable bureau records, yet they leave traces through mobile devices, wallets, repayment channels, merchant payments and contact networks. Coordinated applications, synthetic identities, mule accounts, and repayment manipulation are evident through repeated links across applications and transactions.

The research aim is to develop an analytical model for applying Graph Neural Networks to detect fraudulent transaction patterns in digital lending systems. The study pursues three objectives. The first objective is to define how lending and transaction data should be represented as a heterogeneous temporal graph. The second objective is to explain how GNN architectures identify coordinated and disguised fraud under class imbalance and delayed labels. The third objective is to formulate decision-governance rules that connect graph-based fraud scores with credit workflows, review queues, and monitoring indicators.

The novelty lies in shifting the article from general fraud scoring to the anatomy of graph-based fraud evidence. The proposed view treats a loan application as one event inside a network of borrowers, devices, wallets, repayment channels, merchants and contact links.

The hypothesis states that GNN-based modelling improves fraud-pattern visibility when the lender preserves node roles, relation direction, edge timing and class-imbalance controls before the score enters any approval or review route.

MATERIALS AND METHODOLOGY

The material corpus consists of ten recent publications on graph-based fraud detection, financial graph anomaly detection, consumer loan fraud, online credit loan risk, and credit card transaction networks. The selection retained studies that addressed financial fraud, lending risk or transaction anomaly detection through graph representation, GNN architecture, temporal graph modeling, heterogeneous relations, imbalanced learning, or semi-supervised risk prediction. The corpus covers four research zones: graph construction for financial anomaly detection [4], GNN taxonomies and research gaps in financial fraud detection [2; 6], loan and online credit services [7; 10], and model-level solutions for transaction relations, ranking quality, camouflage, temporal context, and class imbalance [1; 3; 5; 8; 9].

The study uses comparative analysis to distinguish graph construction options, source analysis to identify technical limits, conceptual synthesis to connect GNN fraud detection with digital lending decisions, typologization to group fraud patterns by graph structure, and analytical generalization to formulate an implementation model for a non-experimental article.

RESULTS AND DISCUSSION

Recent research treats financial fraud as a relational detection problem. Reviews of GNN-based financial fraud detection report that researchers most often use supervised and semi-supervised models, while unsupervised, edge-level, and graph-level anomaly detection receive less attention [6]. A later review places the same issue in a broader financial context: GNN models capture relations and time-varying links that are lost when analysts reduce a transaction to a flat row [2]. Digital lending inherits this problem in a sharper form. A fraudulent loan application may have ordinary declared income, a plausible device fingerprint, and no prior default. The suspicious signal appears only after the system compares the applicant with other nodes in the lending graph.

The first analytical result concerns data representation. A digital lender should model borrowers, applications, devices, cards, bank accounts, wallets, merchants, contacts, and repayment channels as different node types. Edges then record relations such as device use, application origin, fund transfer, repayment route, shared account or listed contact. DGraph research supports this direction through a large, dynamic financial graph with about three million nodes, four million dynamic edges, and one million labeled nodes, where anomalous users differ from normal users in structure, neighbor distributions, and temporal behavior [4]. For lending, this finding means that the system should read both applicant attributes and the applicant's network position.

A lending graph needs more than a list of connected entities. Each relation should preserve the reason for its existence. A device edge records access infrastructure. A repayment-account edge records financial execution. A merchant edge records post-disbursement spending. A contact edge records declared social proximity. These relations create different fraud meanings even when they connect the same borrower node to another node. The main design error in digital lending fraud graphs is the compression of all links into one generic neighborhood. Such compression makes the model blind to the difference between a family contact, a repeated device and a suspected mule-account path.

For this reason, the graph layer should be built as a fraud-evidence map. The map links each graph object to a concrete fraud question: whether the applicant shares infrastructure with previous suspicious applications, whether disbursed funds travel through repeated counterparties, whether declared contacts form circular clusters, and whether repayment channels appear across unrelated borrower identities. Table 1 sets out the proposed typology.

Table 1: Typology of graph evidence for fraudulent transaction pattern detection in digital lending

Graph evidence class	Typical nodes and edges	Fraud pattern indicated	Interpretation rule
Identity-infrastructure reuse	Borrower, device, SIM, IP range, application channel	Repeated applications under different identities	Treat as risk only when reuse crosses unrelated borrower profiles or short application windows
Repayment-channel overlap	Borrower, bank account, wallet, card, repayment route	Mule-account proximity or controlled repayment behavior	Distinguish household repayment support from repeated use across unrelated borrowers
Post-disbursement transfer chain	Borrower, wallet, merchant, counterparty, cash-out route	Rapid diversion of loan funds after approval	Read edge timing together with transfer direction and counterparty repetition
Contact-network circularity	Borrower, phone contact, guarantor, emergency contact	Coordinated borrower group or recycled social references	Compare declared relationship type with cluster density and prior fraud proximity
Merchant or agent concentration	Borrower, merchant, agent, payment point	Collusive merchant or cash-out concentration	Treat repeated merchant paths as suspicious only when joined with abnormal timing or shared devices
Label-proximity signal	Applicant node, confirmed fraud node, shared edge path	Hidden connection with known fraudulent behavior	Limit propagation by edge type, distance and recency to reduce false positives

Consumer loan fraud research gives the closest lending-specific support. Xu et al. propose a GNN with a role-constrained conditional random field for consumer loan fraud detection and validate it in a large-scale auto-loan scenario [10]. Their model uses multiple relationship types and constrains users with the same role to have similar representations. This matters for digital lending because a borrower, device, dealer, repayment account, and emergency contact carry different evidentiary meanings. A graph model that blends all neighbors into one undifferentiated signal risks treating ordinary household links and organized fraud rings alike.

Graph construction should therefore preserve role semantics. A shared device between two household members deserves different treatment from a device connected to dozens of applicants with unrelated names and repayment accounts. A merchant link after loan disbursement differs from an emergency-contact link during onboarding. The graph schema must encode these distinctions before the model learns from the data.

Figure 1 translates the typology into a processing sequence. The figure shows a fraud-evidence route: from raw borrower events to typed graph objects, from graph objects to GNN representations, and from representations to a relation-level fraud signal [2; 4; 7; 10].

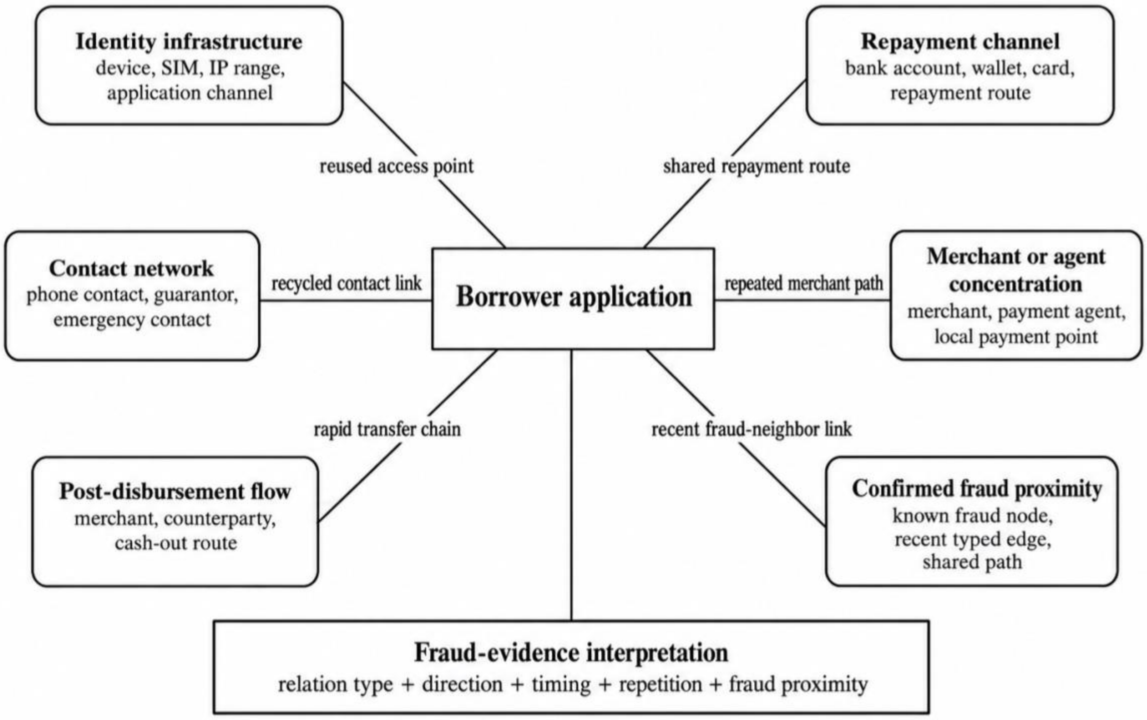

Figure 1. Relational fraud-evidence graph for digital lending applications

The figure separates three operations that are often merged in digital lending platforms. The first operation creates graph objects from borrower events. The second operation lets the GNN learn from typed neighborhoods, temporal edges and minority-class examples. The third operation converts the learned representation into a fraud-pattern signal.

The second analytical result concerns pattern recognition under class imbalance. Fraud cases form a small minority in lending datasets, and this minority often changes its behavior after lenders adjust verification rules. AUC-oriented GNN research addresses this problem by optimizing ranking quality, a design suited to fraud teams that need high-risk cases near the top of a review queue [3]. Accuracy gives weak guidance in this setting because a classifier can look strong while still missing the rare events that investigators care about. Precision at top-k, recall in the highest-risk band and review yield better reflect lending operations.

Imbalance-oriented GNN research reaches the same issue from model design. Tong and Shen propose HHLN-GNN, a fraud detector that uses subgraph generators and neighborhood samplers to handle category imbalance in financial transaction data [8]. Wen et al. propose graph tran-smote, where a subgraph neural network extractor and transformer-based integration address inadequate representation of minority fraud nodes [9]. For digital lending, these studies support a restrictive rule: dense connectivity alone should not trigger fraud classification. Borrowers in underbanked segments may share phones, addresses, workplaces, and local merchants for ordinary economic reasons. The model should seek suspicious relational structure under scarcity.

The third analytical result concerns coordinated fraud. Researchers use GNNs because message passing enables risk information to propagate across connected nodes. In digital lending, a borrower with clean direct attributes may receive a higher fraud score if the graph links the borrower to a device, account, or wallet that is already near-confirmed fraud. This mechanism helps detect synthetic identity chains, mule-account proximity, repeated device use, and organized application clusters. Credit card fraud research by Cherif et al. supports this logic through an encoder-decoder GNN that models relationships between customers and merchants and reports improvements over several comparison models in precision, recall, and F1 score [1]. Although card fraud differs from loan fraud, the relational principle transfers to lending transactions after disbursement.

Figure 2 keeps the processing sequence to connect typed graph relations with representation learning and a fraud-pattern score.

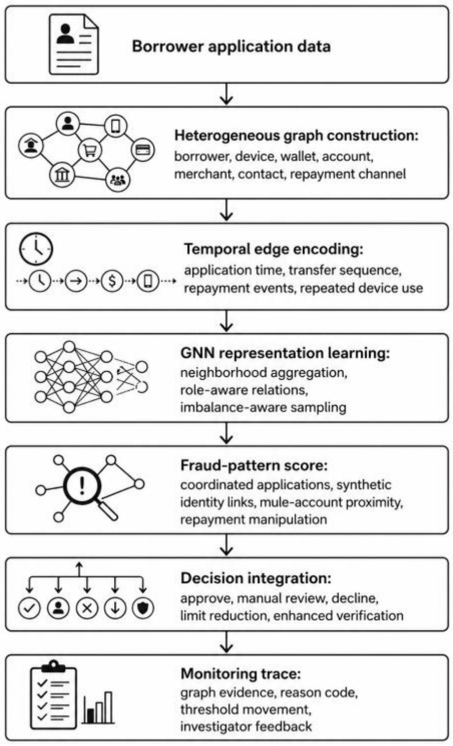

Figure 2. Graph-based GNN layer for fraudulent transaction pattern detection in digital lending systems [2; 4; 7; 10]

Figure 2 separates graph construction from graph learning. The first stage fixes the evidentiary structure: borrower, device, wallet, account, merchant, contact and repayment channel. The second stage lets the GNN aggregate typed neighborhoods with role-aware and imbalance-aware constraints. The output is a fraud-pattern signal tied to coordinated applications, synthetic identity links, mule-account proximity and repayment manipulation.

A comparison across four sources clarifies the operational consequence. DGraph research shows that anomalous financial users differ by graph structure and temporal dynamics [4]. Consumer loan fraud research shows that role information shapes applicant representation [10]. Imbalance-oriented models show that sampling and representation learning must protect minority fraud signals [8; 9]. Together, these studies lead to a lending-specific modeling requirement: the graph schema, role constraints, and imbalance treatment should be designed as a single unit. If engineers build only a dense graph, ordinary social proximity can distort the score. If they add role semantics without imbalance control, rare schemes can remain hidden. If they add imbalance control without temporal edges, short-lived coordinated attacks can pass through the system before labels arrive.

The fourth analytical result concerns temporal context and camouflage. Fraudsters can change device use, transfer routes, repayment timing, and identity combinations after a lender changes controls. Lou et al. propose a context-encoding and adaptive-aggregation fraud detector that encodes time intervals of outgoing edges and out-degree, then adjusts aggregation to address camouflage behavior [5]. This finding fits with digital lending because repeated applications within a short time window have a a different meaning from slow, ordinary reuse within a household. Temporal frequency, edge direction, and degree patterns help separate common borrower behavior from coordinated fraud operations.

Online credit loan services add label scarcity to the temporal problem. Tang et al. propose a Snorkel-based semi-supervised GNN for online credit loan risk prediction, where graph-based weak supervision helps address limited labels in anti-fraud scenarios [7]. Digital lenders face the same delay. Confirmed fraud labels may appear after non-payment, a chargeback, an identity dispute, a manual investigation, or law-enforcement feedback. During that delay, a purely supervised model trains on incomplete ground truth. Weak supervision and semi-supervised learning can reduce this gap if the lender records expert rules, investigator decisions, and later outcomes in a controlled feedback process.

The fifth analytical result concerns explainability. Reviews of financial fraud GNNs identify deployment, interpretability, and operational integration as persistent gaps [2; 6]. For digital lending, the decision team needs a usable trace: which relation type raised risk, how recent the relation was, how close the applicant was to confirmed fraud, and which policy threshold changed the final route. A borrower-facing adverse action notice may require simplified reason codes, whereas an internal audit file requires relation-level evidence and model version history.

The sixth analytical result concerns real-time implementation. Digital lending engines require short decision cycles, while full graph computation may be expensive for every application. The literature supports a hybrid architecture. Engineers can refresh stable graph embeddings on a scheduled basis and score recent high-velocity edges during the synchronous decision path. DGraph demonstrates the relevance of large dynamic financial graphs for anomaly detection [4]. Context-aware GNN research shows that edge timing and out-degree should be incorporated into the model when fraud behavior changes over time [5]. A lending service can therefore call precomputed borrower, device, or account embeddings, enrich them with recent edge events, and apply a calibrated fraud threshold.

The seventh analytical result concerns underbanked borrowers. Alternative data can support credit access when bureau histories are thin, yet relational fraud controls can penalize legitimate users if the model treats shared devices or contacts as automatic red flags. Consumer loan fraud research points to role constraints [10], while online credit loan research points to semi-supervised learning under scarce labels [7]. In practice, a shared contact should receive meaning through role, frequency, recency, and proximity to confirmed fraud. This approach reduces false positives in communities where people share phones, use payment agents, or use family repayment channels.

The final analytical result concerns the boundary between graph evidence and borrower treatment. A lender should avoid automatic decline based only on network proximity. Graph proximity has evidentiary value when it appears together with relation type, recency, repetition and known fraud labels. Weak proximity should move an application to review or enhanced verification. This distinction is especially relevant for thin-file borrowers, because shared phones, family repayment routes and local payment agents can reflect ordinary access conditions.

The implementation problem in this article concerns the evidentiary meaning of graph relations. A lender can produce an accurate-looking fraud score and still fail operationally if investigators cannot see which relation created suspicion. A GNN layer therefore needs a relation-to-action rule. The rule translates graph evidence into a limited set of review actions: enhanced identity check, repayment-channel verification, merchant-path review, manual fraud investigation or score suppression when the relation is too weak.

This design differs from latency architecture. Latency work asks where the system spends time. The GNN fraud article asks which relational signal deserves attention. It also differs from CI/CD work, where the central question concerns model promotion and rollback. Here the central question concerns the evidentiary threshold between an ordinary shared link and a suspicious coordinated pattern.

Model selection should follow graph design. A team can choose GCN, GAT, heterogeneous GNN, temporal GNN, or a hybrid embedding pipeline only after it defines node types, edge direction, retention periods, and legally usable signals. A technically advanced architecture will not correct a graph that confuses a family contact with a merchant relation or a repayment account with a device fingerprint.

The decision logic should separate credit risk and fraud risk. A borrower may have weak repayment capacity without suspicious graph links. Another applicant may appear creditworthy while the network shows proximity to confirmed fraud. The lending engine should keep these situations distinct through separate score bands, review routes, and reason codes. This design provides underwriters with clearer files and gives compliance staff a trace that links the decision to a specific relation type.

Table 2 compares the layers needed for GNN-based fraud detection in digital lending. The table separates graph engineering, model learning, and operational use, since these functions often merge too early during AI adoption.

Table 2: Relation-to-action rules for graph-based fraud alerts

Fraud-evidence signal	Minimum supporting condition	Recommended decision route	Reason recorded in trace
Repeated device across unrelated applicants	Device appears across unrelated borrower profiles within a short application window	Enhanced identity verification	Device reuse across unrelated applications
Shared repayment account	Same repayment route appears across unrelated borrower identities	Manual review before approval	Repayment-channel overlap
Rapid post-disbursement transfer chain	Funds move through repeated counterparties shortly after loan release	Limit reduction or fraud review	Post-disbursement transfer concentration
Dense recycled contact cluster	Several applicants list overlapping contacts without declared household relation	Manual review	Contact-network circularity
Proximity to confirmed fraud node	Applicant connects to confirmed fraud through recent typed edges	Fraud investigation queue	Recent graph proximity to confirmed fraud
Weak shared-link evidence	Only one low-risk relation appears without temporal concentration	Suppress adverse action from graph score alone	Insufficient relational support

A lender may build or buy a strong GNN model, yet still receive little operational value if the fraud score is entered into the decision engine as an unexplained number. The system needs relation-level traces, policy thresholds, and feedback labels from investigators. Without them, the model may improve detection metrics while making it harder to defend against adverse decisions.

Monitoring should cover predictive quality and workflow pressure. A model that captures more fraud but floods manual review queues will slow approvals. A model that lowers review volume but increases false declines will damage access for legitimate borrowers. The monitoring design should connect model outputs with business, compliance, and inclusion indicators.

The monitoring framework moves evaluation beyond model accuracy. Fraud detection in lending affects credit access, user friction, investigator workload, and dispute handling. A graph model should therefore enter production through score bands, manual-review thresholds and controlled feedback loops. Confirmed fraud labels should update the graph, but governance checks should prevent one burst of activity from distorting the entire network.

A defensible implementation sequence can follow five stages. The first stage builds a minimal graph from high-confidence relations: borrower, application, device, repayment account and phone number. The second stage adds temporal edges for application sequence, repayment behavior and post-disbursement movement. The third stage pilots an imbalance-aware GNN against confirmed fraud labels and investigator outcomes. The fourth stage introduces score bands into the lending engine, starting with manual review before automatic decline. The fifth stage adds monitoring, threshold adjustment, and explanation traces.

Underbanked segments require tighter safeguards. Shared devices, family contacts, and dense local networks can reflect ordinary borrowing conditions. The graph layer should read these patterns through relation type, frequency, recency, and proximity to confirmed fraud. A simple shared-link penalty would create avoidable false positives and reduce the value of alternative data for inclusion.

CONCLUSION

Digital lending fraud detection benefits from graph representation when the lender treats each application as part of a typed relational structure. Borrowers, devices, wallets, accounts, merchants, contacts and repayment channels should not enter the model as interchangeable neighbors. Their evidentiary meaning depends on role, edge direction, timing, repetition and proximity to confirmed fraud.

GNN architectures support coordinated-fraud recognition because message passing can expose hidden links among applicants, devices, repayment routes and counterparties. This advantage becomes usable only when the model protects rare fraud signals through imbalance-aware learning and preserves temporal information during aggregation.

The proposed analytical model confirms the hypothesis at the conceptual level. GNN-based modelling improves visibility into fraudulent transaction patterns when the graph is designed as an evidence structure. The practical result is a narrower and more defensible fraud layer: it identifies suspicious relational patterns, records the relation behind the alert and leaves final borrower treatment to controlled review rules.

REFERENCES

1. Cherif, A., Ammar, H., Kalkatawi, M., Alshehri, S., & Imine, A. (2024). Encoder-decoder graph neural network for credit card fraud detection. *Journal of King Saud University - Computer and Information Sciences*, *36*(3), Article 102003. https://doi.org/10.1016/j.jksuci.2024.102003
2. Cheng, D., Zou, Y., Xiang, S., & Jiang, C. (2025). Graph neural networks for financial fraud detection: A review. *Frontiers of Computer Science.* https://doi.org/10.1007/s11704-024-40474-y
3. Huang, M., Liu, Y., Ao, X., Li, K., Chi, J., Feng, J., Yang, H., & He, Q. (2022). AUC-oriented graph neural network for fraud detection. In Proceedings of the ACM Web Conference 2022 (pp. 1311-1321). *Association for Computing Machinery.* https://doi.org/10.1145/3485447.3512178
4. Huang, X., Yang, Y., Wang, Y., Wang, C., Zhang, Z., Xu, J., Chen, L., & Vazirgiannis, M. (2022). DGraph: A large-scale financial dataset for graph anomaly detection. *Advances in Neural Information Processing Systems, 35*, 22765-22777.
5. Lou, C., Wang, Y., Li, J., Qian, Y., & Li, X. (2025). Graph neural network for fraud detection via context encoding and adaptive aggregation. *Expert Systems with Applications, 261*, Article 125473. https://doi.org/10.1016/j.eswa.2024.125473
6. Motie, S., & Raahemi, B. (2024). Financial fraud detection using graph neural networks: A systematic review. *Expert Systems with Applications, 240*, Article 122156. https://doi.org/10.1016/j.eswa.2023.122156
7. Tang, H., Wang, C., Zheng, J., & Jiang, C. (2024). Enabling graph neural networks for semi-supervised risk prediction in online credit loan services. *ACM Transactions on Intelligent Systems and Technology, 15*(1), Article 13, 1-24. https://doi.org/10.1145/3623401
8. Tong, G., & Shen, J. (2023). Financial transaction fraud detector based on imbalance learning and graph neural network. *Applied Soft Computing, 149*, Article 110984. https://doi.org/10.1016/j.asoc.2023.110984
9. Wen, J., Tang, X., & Lu, J. (2024). An imbalanced learning method based on graph tran-smote for fraud detection. *Scientific Reports*, *14*, Article 16560. https://doi.org/10.1038/s41598-024-67550-4
10. Xu, B., Shen, H., Sun, B., An, R., Cao, Q., & Cheng, X. (2021). Towards consumer loan fraud detection: Graph neural networks with role-constrained conditional random field. *Proceedings of the AAAI Conference on Artificial Intelligence, 35*(5), 4537-4545. https://doi.org/10.1609/aaai.v35i5.16582