Computer Fraud and Security
ISSN (online): 1873-7056

Transformer-Based Anomaly Detection for First-Party Fraud Patterns Across Transaction Graphs

Gautham Paspala

Independent Researcher, USA

Abstract

First-party fraud, or friendly fraud, is a new and emerging problem in digital commerce. It occurs when the legitimate cardholder takes advantage of chargebacks to obtain a refund for goods or services, despite having already received them. Because of the time between transaction and the filing of a fraudulent dispute, point of transaction detection does not work. Simple rule systems and basic machine learning models that don't compare one account to others or rely on fixed features are not good enough for understanding the complex behaviors and connections involved in more advanced types of friendly fraud. Heterogeneous temporal graphs can be formed from these transactions using graph neural networks and transformer-based attention architectures. The heterogeneous message-passing layers, i.e. the graph attention layer and the multi-head temporal self-attention layers surface cross-account, cross-device, across-time, and cross-dimension interaction signals of fraud missing from standard detection. The use of fused structural and sequential representations improves detection efficacy across opportunistic, habitual, and organized fraud typologies. Cost-sensitive threshold calibration and human-in-the-loop review allocation enable deployment in production without excessive burdens of false positives. .

Keywords: Graph Neural Networks, Transformer Attention Mechanisms, First-Party Fraud Detection, Temporal Transaction Graphs, Chargeback Abuse

1. THE EVOLVING THREAT LANDSCAPE OF FIRST-PARTY FRAUD IN DIGITAL PAYMENTS

First-party fraud, or "friendly fraud," has been referred to as one of the most expensive and difficult forms of fraud to combat in the digital commerce industry. First-party fraud differs from third-party fraud, in which a criminal compromises a legitimate account via stolen credentials, phishing, or account takeover. First-party fraud occurs when the actual account owner initiates the transaction but then proceeds to get refunds for legitimate goods or services by using chargeback processes. The fraud here is that it is a dispute of the transaction, which is what renders point-of-transaction detection architectures largely ineffective [1] .

The taxonomy of first-party fraud behavior runs from opportunistic abuse, the least complicated and least intentional type of fraud (and the most common), through fraudsters who purchased legitimately but then, as a result of buyer remorse, ambiguous delivery conditions, or unauthorized use by others within the same household, abuse the chargeback mechanism. Habitual fraudsters are a more analytically tractable category of chargeback abuser but still represent an important portion of friendly fraud. They also engage in a pattern of purchase disputes, using patterns in their behavior (i.e., frequency and merchants involved) to remain undetected. Organized fraudsters at the highest point on the sophistication spectrum engage in organized schemes using coordinated campaigns to exploit promotional offers, return policies, and chargeback windows across a merchant's portfolio [2].

Fraud losses involving credit cards are enormous. Global credit card fraud losses exceeded $32 billion in 2022 alone. In Canada, over 91,190 fraud incidents happened with a loss of $531 million to 57,055 people, with reported cases showing the limits of rule-based systems [2]. Thresholds have the effect of creating boundaries that are easily breached by expert adversaries. Static rules consider features in isolation rather than jointly, which prevents rules from covering much of the first-party fraud volume known to be valid. [1] .

Machine learning models based on classical algorithms (logistic regression, random forests, gradient increasing) are more successful than rule-based approaches, but these too are hampered by data sparsity, high dimensionality

and nonlinearity, among other issues in financial fraud detection [2]. Emerging patterns of fraud exploit behavioral dimensions not captured by existing datasets, eluding detection and leading to reactive models that are always behind evolving fraud patterns .

Graph neural network (GNN) and transformer architecture models have been proposed to address these issues. A systematic literature review with 388 distinct papers from five academic databases (Web of Science, Scopus, IEEE Xplore, SpringerLink and ACM) has shown that GNNs are a new promising approach for financial fraud detection in complex networks, with 33 high-quality identified papers confirming a comparative advantage over customary approaches [2]. GNNs model the entire transaction ecosystem as graphs, enabling them to reason about multi-hop relationships between entities, while transformer attention mechanisms are effective at modeling complex and non-linear temporal patterns, increasing performance over previous methods .

2. GRAPH-BASED MODELING OF TRANSACTION ECOSYSTEMS

When set up as a mixed-time graph, payment transaction data shows important structural details that aren't visible in the usual tables or lists of payment transactions. The heterogeneous temporal graph includes nodes corresponding to the different entities in the payment ecosystem, such as cardholder accounts, merchant identifiers, device fingerprints, transaction events, shipping addresses, IP addresses, and contact IDs. The edges connecting the entities reflect the semantic relationships between the entities. Every edge has a timestamp and a set of attributes that provide detailed information about the nature of the relationship. This model captures the entire web of relationships between the entities of the payment network [3] .

The high expressiveness of this graph representation is due to the fact that the one-hop neighborhood of a cardholder node contains the behavioral profile of the cardholder, including his or her merchants, devices, and number of transactions (or transaction frequency), across all devices and merchants. In particular, the two-hop neighborhood of a cardholder contains far more diagnostic links: other cardholders sharing devices, merchants frequently visited by cardholders with similar behavioral profiles, and addresses receiving shipments for multiple ostensibly unrelated accounts. Fraud patterns, especially of organized fraud rings, are often manifested not in individual node attributes, but in the larger structural properties of the extended neighborhood [4] .

Experiments for benchmarking have also verified the importance and feasibility of this structure. The DGraph data set, which the Finvolution Group used for detecting financial fraud, includes 3,700,550 users and 4,300,999 directed edges, and it mimics the size and complexity of real financial networks found in actual use cases. The experiments on the dataset reveal that the TGNs outperform the static graph baselines, with the best Test AUC score obtained for the TGNs being 0.7747 compared to 0.6829 for the best hypergraph baseline and 0.6507 for the standard (non-temporal) MLP models (an absolute improvement of 13.18%) [3] .

To address this problem, the temporal dimension of the graph can be encoded by timestamping each edge with a tuple containing the source vertex, the destination vertex, and the timestamp e = (u_s, u_d, t) [3]. This allows requesting data from arbitrary time windows and enables models to learn temporal patterns at different time scales, such as transaction velocity over a one-hour window or frequency of disputes over a time period of several months.

Graph construction can often be combined with additional behavior signals. GNNs, as noted in over 4791 works, can refer to message-passing algorithms that move messages along links and learn representations of the graph initializations and relationships between nodes in constructed graphs [4]. These features convert the raw event data into dense attribute vectors suitable for processing by a graph neural network, enabling superior detection performance compared to previous architectures .

Metric / Parameter	Value	Description
Total Users in DGraph Dataset	3,700,550	Number of user nodes in the benchmark dataset
Total Directed Edges in DGraph Dataset	4,300,999	Number of directed edges representing relationships

Best Test AUC – Temporal Graph Networks (TGN)	0.7747	Highest AUC score achieved by TGN models
Best Test AUC – Hypergraph Baseline	0.6829	Best AUC score from hypergraph baseline models
Best Test AUC – Standard MLP Models	0.6507	AUC score for non-temporal MLP baseline models
Absolute AUC Improvement (TGN vs. MLP)	13.18%	Performance gain of TGN over standard MLP models

Table 1: Quantitative Benchmarks for Graph-Based Fraud Detection Models on the DGraph Dataset [3, 4]

3. GRAPH NEURAL NETWORK ARCHITECTURE FOR FRAUD REPRESENTATION LEARNING

This model employs a graph neural network with multiple layers of message-passing, which produces richer node embeddings from neighboring nodes. For each layer of passing, the old node embedding is aggregated with the rest of the neighboring nodes' embeddings into a new embedding vector, which iteratively gathers structural information passed through its higher-order neighborhoods. After multiple layers, the node embedding can contain not only information about the node that it is representing, but also the statistical properties of the neighborhood, providing a richer representation for fraud compared to the account level [5]. .

The heterogeneous and multi-level structure of transaction graphs requires extensions to customary homogeneous GNNs. The base model is a GNN that maps a graph and one of its nodes to an n-dimensional Euclidean space and updates their states in an iterative manner. Empirical evidence demonstrates that the procedure converges quickly, requiring only 5 to 15 iterations to reach a fixed point [6]. A type-specific parameterization follows the intuition that the semantic information of relations is different, e.g. fraud-relevant information is in a connection to another device, while a relationship connection to a merchant contains different information [5]. .

In GATs, the message-passing framework is extended in that the neighboring nodes are adaptively weighted according to their importance for the prediction task. Rather than equal weights, attention coefficients are computed for neighboring nodes, which are normalized to focus the feature aggregation on the most relevant nodes. The GAT architecture leverages multi-head attention, consisting of K independent attention heads. In the case of transductive tasks, K = 8 attention heads, each producing F = 8 features for a total of 64 features per node, and for inductive tasks K = 4 attention heads each producing F = 256 features, for 1024 features total per layer. On four standard datasets, including the Cora citation network (2,708 nodes, 5,429 edges) and a protein-protein interaction dataset (56,944 nodes, 818,716 edges), GAT achieved a micro-averaged F1 score of 0.973 on inductive tasks, which is 20.5% higher than the best GraphSAGE performance [5]. .

The depth of the graph neural network controls the size of the receptive field. Three or four graph convolution layers provide an ideal compromise depth, providing sufficiently deep propagation of information through the network to capture organized patterns of fraud, with sufficient specificity for the embedding. One GAT attention head that computes F features has a time complexity of $O(|V|FF' + |E|F)$ (the same as the baseline GCN method) [5]. To retain the importance of single features, a skip connection is applied between GAT layers. This prevents the dilution of individual features, which are critical in scoring nodes for individual fraudulent activity [6] .

Metric / Parameter	Value	Description
Message-Passing Iterations to Convergence	5–15	Empirically sufficient iterations to reach a fixed point
Attention Heads—Transductive Tasks (K)	8	Number of independent attention heads for transductive settings

Features per Head – Transductive Tasks (F)	8	Features produced per attention head in transductive tasks
Total Features per Node – Transductive Tasks	64	K × F = 8 × 8 total features per node
Attention Heads—Inductive Tasks (K)	4	Number of independent attention heads for inductive settings
Features per Head – Inductive Tasks (F)	256	Features produced per attention head in inductive tasks
Total Features per Layer – Inductive Tasks	1,024	K × F = 4 × 256 total features per layer
Optimal Graph Convolution Layers	3–4	Ideal depth for capturing organized fraud patterns
GAT Micro-Averaged F1 Score (Inductive)	0.973	Achieved on inductive benchmark tasks
Improvement over Best GraphSAGE	20.50%	GAT F1 improvement relative to GraphSAGE baseline
Cora Citation Network – Nodes	2,708	Number of nodes in the Cora benchmark dataset
Cora Citation Network – Edges	5,429	Number of edges in the Cora benchmark dataset
Protein-Protein Interaction Dataset – Nodes	56,944	Number of nodes in the PPI benchmark dataset
Protein-Protein Interaction Dataset – Edges	818,716	Number of edges in the PPI benchmark dataset
Standard Benchmark Datasets Used	4	Total datasets used for GAT evaluation

Table 2: Architectural Parameters and Performance Metrics of Graph Attention Networks (GAT) for Fraud Detection [5, 6]

4. TRANSFORMER ARCHITECTURES AND ATTENTION MECHANISMS FOR BEHAVIORAL ANOMALY DETECTION

In parallel, businesses use a transformer on the transaction time series of the cardholder. This allows us to detect temporal behavioral patterns that cannot be captured at the graph level because they involve complex non-local dependencies that cannot be preserved in convolutional or recurrent networks with sliding windows of limited past context. In addition, a self-attention layer only requires O(1) sequential operations to encode all positions, whereas a recurrent layer requires O(n) operations to evaluate as many sequential time steps, yielding a speed-up when sharing long behavioral histories [8] .

The transaction sequences are formed by sorting purchase events by their timestamp, which are represented by dense vectors of features. Dispute events are treated as special transaction types at their filed timestamp, enabling the model to learn the signature behavior of the period before the fraud dispute. The best-performing model, the Hybrid Temporal Attention-Gated Model (HTAGM), achieved an F1-score of 0.90 and a ROC-AUC score of 0.99, outperforming the single LSTM baseline (0.80 F1, 0.96 ROC-AUC) and the simple GRU baseline (0.82 F1, 0.97 ROC-AUC) by 5-12% in terms of F1-score. As a result, the hybrid temporal attention architectures were shown to outperform recurrent methods [7]. .

Positional encoding builds on the positional encoding used in the Transformer model to work with irregular time steps in actual transaction histories. These encodings are calculated by applying the sine and cosine functions to

frequencies scaled to a d_model = 512-dimensional space and whose wavelengths are spaced geometrically between 2π and 10,000 · 2π [8]. Continuous-time versions of this encoding family would allow the model to discriminate between a transaction history with frequent and infrequent transactions, a potential indicator of certain types of fraud .

The multi-head self-attention mechanism allows learning attention patterns in parallel. The Transformer uses h = 8 parallel heads with d_k = d_v d_model/h = 64. This is as computationally expensive as the single attention layer with full dimensions (d_k = d_v = d_model), which was used in the seminal attention paper [8]. In HTAGM, each transaction is weighted by the temporal attention module in the fraud detection task to pay attention to abnormal spikes in spending, irregular time intervals between spending, and anomalous interactions with merchants. The GRU/LSTM unit selectively passes past information forward and stores it as long-term memory and acts as a gate to filter noise [7].. .

Cross-modal fusion takes the structured embeddings from the GNN and sequential embeddings from the transformer through cross-attention. The model outperformed baselines including logistic regression (0.91), random forest (0.95), and support vector machines (SVM) (0.93), with an overall accuracy of 0.982 [7]. Multi-task learning with auxiliary prediction targets helped to learn better representations of behavior in addition to fraud classification.

Metric / Parameter	Value	Description
Sequential Operations – Self-Attention Layer	O(1)	Operations required to encode all positions
Sequential Operations – Recurrent Layer	O(n)	Operations required to evaluate sequential time steps
HTAGM – F1-Score	0.9	Best-performing model F1 score
HTAGM – ROC-AUC Score	0.99	Best performing model ROC-AUC score
LSTM Baseline – F1-Score	0.8	Single LSTM baseline F1 score
LSTM Baseline – ROC-AUC Score	0.96	Single LSTM baseline ROC-AUC score
GRU Baseline – F1-Score	0.82	Simple GRU baseline F1 score
GRU Baseline – ROC-AUC Score	0.97	Simple GRU baseline ROC-AUC score
HTAGM Improvement over Baselines (F1)	5–12%	F1-score improvement over recurrent baselines
Positional Encoding Dimension (d_model)	512	Dimensional space for sine/cosine positional encoding
Wavelength Range – Lower Bound	2π	Minimum wavelength for positional encoding frequencies
Wavelength Range – Upper Bound	10,000 · 2π	Maximum wavelength for positional encoding frequencies
Parallel Attention Heads (h)	8	Number of parallel heads in multi-head self-attention
Key/Value Dimensions (d_k = d_v)	64	d_model / h = 512 / 8 per attention head
Overall Model Accuracy	0.982	Cross-modal fusion model accuracy

Logistic Regression Accuracy	0.91	Baseline comparison model accuracy
Random Forest Accuracy	0.95	Baseline comparison model accuracy
SVM Accuracy	0.93	Baseline comparison model accuracy

Table 3: Performance Comparison of Transformer-Based and Baseline Models for Fraud Behavioral Anomaly Detection [7, 8]

5. BALANCING DETECTION PRECISION AGAINST FALSE POSITIVE COSTS IN PRODUCTION SYSTEMS

Deploying fraud detection models in production needs to consider the asymmetric cost structure. In the case of friendly fraud detection, false positives incur costs from managing analysts for investigations and losing customers, in addition to the costs of the investigation itself. The average credit card relationship, which can be valued over its lifetime, means that the loss from false positives can exceed the loss from fraud.

The ROC curve plots the tradeoff between the level of fraud detected and the rate of false positives at each possible threshold. The area under the curve (AUC) metric summarizes discrimination performance but is not useful for operational threshold determination. The cost-sensitive threshold optimization thus addresses the tradeoff by introducing the cost of each type of outcome as a factor. Cost-sensitive models have been shown to yield important improvement over raw-feature baselines. For example, using 236,735 transactions with a 1.50% fraud rate and 895,154 Euros total fraud losses, savings increased by 201% on average with cost-sensitive models and by 287% with combined feature strategies [9] .

Tiered thresholding extends single thresholding by linking interventions to confidence of detection. If fraud scores are highly confident, action is immediately taken at the account level. If scores are moderately confident, they will be queued for investigation; if ratings are low, they will be monitored. The addition of the aggregated and temporal behavior features improved savings by 16.4% and 13%, respectively [9]. Each tier forms a separate operating point along the precision-recall curve, so the parameters of each must be tuned differently, as they have different implications on the underlying cost functions [10] .

As friendly fraud varies across segments, global thresholds produce many false positives in low-fraud segments and few detections in high-fraud segments. Segment-specific calibration circumvents this heterogeneity. Fairness-aware calibration requires the added constraint that the change to the threshold needs to maintain an equitable false positive rate across demographic groups, as required by fair lending [10].

Human-in-the-loop review optimization addresses feasibility considerations regarding fully automated decisioning. For instance, review prioritization of investigations seeks to optimize human analysts' potential fraud recovery based on the expected value of fraud, model confidence, amount of evidence available, and time-sensitive requirements or constraints for impacting the fraud. Analysts provide highly detailed, labeled training data that cannot be observed from model output, and systematized study of overrides identifies weaknesses that inform architectural redesign [13].

Metric / Parameter	Value	Description
Total Transactions in Dataset	236,735	Number of transactions used in cost-sensitive model evaluation
Fraud Rate	1.50%	Proportion of fraudulent transactions in the dataset
Total Fraud Losses	895,154 Euros	Aggregate fraud losses in the evaluation dataset

Savings Improvement – Cost-Sensitive Models	201%	Average savings increase using cost-sensitive models over raw-feature baselines
Savings Improvement – Combined Feature Strategies	287%	Average savings increase using combined feature strategies
Savings Improvement – Aggregated Behavior Features	16.40%	Additional savings from adding aggregated behavior features
Savings Improvement – Temporal Behavior Features	13%	Additional savings from adding temporal behavior features

Table 4: Quantitative Benchmarks for Threshold Optimization Strategies and Feature-Driven Savings in Fraud Detection [9, 10]

6. EXPERIMENTAL EVALUATION, INTERPRETABILITY, AND FUTURE RESEARCH DIRECTIONS

The evaluation of graph-transformer architectures on fraud detection shows both methodological care and reliable performance estimations under realistic deployment conditions. The most important is temporal validation: standard cross-validation splits historical datasets at random and causes leakage from the future to the training set, resulting in optimistically biased estimates with respect to real-world deployment conditions. Rather than randomly picking train/validate splits, temporal validation involves training models exclusively on earlier time periods and validating on later time periods to simulate the real-world task of generalizing to novel future behavioral patterns. Out-of-time testing extends this, beginning validation on successively later time periods to monitor the rate of model degradation to inform refresh cycles [11] .

Rule-based detection systems generally have the lowest coverage across all detection tasks, with increased coverage after applying customary, non-graph-based ML techniques, such as gradient-increased trees, to feature-engineered representations of the input data. Graph-only architectures, which use GNNs without a temporal sequence model, have, in turn, higher coverage. Likewise, transformer-only architectures with behavioral sequences without a graph representation model also have higher coverage. The full graph-transformer fusion architecture achieves the best detection performance in terms of recall and precision. The performance gap between the full graph-transformer fusion architecture and graph- or transformer-only models shows that the two modalities provide different information, whose fusion enables synergistic performance. Ablation studies further show that the graph attention heads are more important to identify organized fraud, while the temporal attention is most important to identify habitual fraud [11] .

Attention maps of true fraud cases retain expert knowledge while revealing new patterns. Attention is always on the most recent period prior to the assessment to capture behavioral drift, the baseline period to detect anomalies, and the timestamps of past disputes (habitual fraud is strongly autocorrelated). The graph attention model shows that device links generally have much higher attention weights than address links, establishing that device-sharing relationships are a superior fraud signal. It also uncovered new relationships in the form of timing signatures that are not explicitly captured by the set of input features. This indicates the potential of the model to discover signals of fraud on its own [12] .

The adversarial robustness analysis describes the detection performance when the fraudsters adapt their attack strategy to avoid detection. The graph-transformer architecture performs considerably better than the customary approaches under moderately adaptive attack scenarios. One explanation for this is that this architecture is mainly driven by structural network patterns, which are more expensive to change, and temporal attention patterns, which capture rather than measure a point in time [12] .

Promising future research directions include multimodal data integration, including natural language processing of claims text and supporting documents; causal inference to separate potential fraud indicators from other correlated structures; continual learning to evolve the model to adapt to the underlying environment; and privacy-preserving models such as federated learning and differential privacy to comply with increasing regulatory and ethical challenges [13,14].

CONCLUSION

Considering the intersection of graph networks and transformer networks, we present a technically sound and deployed approach to first-party fraud detection in a heterogeneous ecosystem of financial transactions. By modeling the cardholder-merchant-device relationship as a temporal graph and leveraging multi-head attention over the ordered behavioral representation of cards, merchants, and devices, this approach is able to observe many fraud signals unavailable to previous methods. Graph attention mechanisms are particularly good at detecting organized fraud networks based on structural device sharing and timing synchronization behaviors, whereas temporal self-attention mechanisms are successful in modeling slowly evolving behavioral sequences, keeping an efficient separation between habitual fraudsters and normal cardholders. Explicit cost-sensitive threshold optimization and segment-specific calibration ensure that improvements in detection performance will translate into improved financial returns from fraud detection, rather than imposing a high false positive burden on honest customers. Furthermore, attention-based interpretability will yield fraud classifications that are interpretable, auditable, and aligned with fairness or regulatory objectives. However, continual developments in continual learning, multimodal information synthesis, and privacy-preserving federated architectures are essential to ensure the longevity and continued advancement of fraud detection across the payment landscape in the context of ever-evolving fraud patterns in adversarial environments.

REFERENCES

[1] Soroor Motie and Bijan Raahemi, "Financial fraud detection using graph neural networks: A systematic review," Expert Systems with Applications, Volume 240, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0957417423026581

[2] Mayank Kale, "Detecting credit card fraud using machine learning," ResearchGate, 2024. [Online]. Available: https://www.researchgate.net/publication/380534576_Detecting_credit_card_fraud_using_machine_learning

[3] Yejin Kim et al., "Temporal Graph Networks for Graph Anomaly Detection in Financial Networks," arXiv:2404.00060v1, 2024. [Online]. Available: https://arxiv.org/pdf/2404.00060

[4] Jie Zhou et al., "Graph neural networks: A review of methods and applications," AI Open, Volume 1, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666651021000012

[5] Petar Velickovi et al., "Graph attention networks," arXiv:1710.10903v3, 2018. [Online]. Available: https://arxiv.org/pdf/1710.10903

[6] Franco Scarselli et al., "The graph neural network model," IEEE Transactions on Neural Networks, Vol. 20, No. 1, 2009. [Online]. Available: https://par.cse.nsysu.edu.tw/resource/paper/2024/240115/The_Graph_Neural_Network_Model.pdf

[7] Ashish Vaswani et al., "Attention Is All You Need," arXiv:1706.03762v7, 2023. [Online]. Available: https://arxiv.org/pdf/1706.03762

[8] M. Dattatreya Goud et al., "A Hybrid Temporal Attention–Gated Model for Extracting Transactional Behavior Patterns in Fraud Detection," American Data Science Journal for Advanced Computations, Volume 3, Issue 4, 2025. [Online]. Available: https://d1wqtxts1xzle7.cloudfront.net/125779379/A_Hybrid_Temporal_Attention-libre.pdf

[9] Clifton Phua et al., "A Comprehensive Survey of Data Mining-based Fraud Detection Research," arxiv. [Online]. Available: https://arxiv.org/pdf/1009.6119

[10] Alejandro Correa Bahnsen et al., "Feature engineering strategies for credit card fraud detection," Expert Systems With Applications, 2016. [Online]. Available: https://albahnsen.github.io/files/Feature%20Engineering%20Strategies%20for%20Credit%20Card%20Fraud%20Detection_published.pdf

[11] Tharindu R. Bandaragoda et al., "Isolation-based anomaly detection using nearest-neighbor ensembles," Wiley, 2016. [Online]. Available: https://www.researchgate.net/profile/Tharindu-Bandaragoda/publication/322359651_Isolation-based_anomaly_detection_using_nearest-neighbor_ensembles_iNNE/links/5e91651092851c2f5294c5ac/Isolation-based-anomaly-detection-using-nearest-neighbor-ensembles-iNNE.pdf

[12] Tianchi Yang et al., "HGAT: Heterogeneous Graph Attention Networks for Semi-supervised Short Text Classification," ACM Transactions on Information Systems (TOIS), Volume 39, Issue 3, 2021. [Online]. Available: https://dl.acm.org/doi/10.1145/3450352

[13] "Transformational leadership and its influence on business growth in the real estate industry," Lex Localis – Journal of Local Self-Government, pp. 1–11, 2023. [Online]. Available: https://doi.org/10.52152/802910

[14] G. Beeyani, "Menu engineering and innovation: Strategies for enhancing customer experience and revenue generation," Review of Contemporary Philosophy, vol. 22, no. 1, pp. 6962–6970, 2023.