Avaliando Modelos de Graph Neural Networks para Detecção de Usuários Fraudulentos em e-Commerce

Larissa de Andrade Silva[1], **Eduardo L. Feitosa**[1]

[1]Instituto de Computação (IComp) - Universidade Federal do Amazonas (UFAM)
Manaus – AM – Brazil

{las,efeitosa}@icomp.ufam.edu.br

Abstract. *Models based on graphs and the combination with deep learning, the Graph Neural Networks (GNN), have been used to detect fraud in electronic commerce with promising results. In this paper, models that use node classification, based on neighborhood information, and community recognition, using real datasets were evaluated. The results, although promising (accuracy ranging from 50% to 86%), show that it is still necessary to study and investigate them better so that they can, in the future, act in anti-fraud solutions.*

Resumo. *Modelos baseados em grafos, Graph Neural Networks ou (GNN), vêm sendo empregados na detecção de fraudes no comércio eletrônico com resultados promissores. Neste trabalho foram avaliados modelos que utilizam classificação sobre os nós, baseados em informações da vizinhança, e o reconhecimento de comunidade, utilizando datasets reais. Os resultados demonstram que, embora promissores (acurácia variando de 50% a 86%), ainda é preciso estudar e investigá-los melhor para que possam, no futuro, atuarem em soluções anti-fraude.*

1. Introdução

Hoje é impossível não dizer que o comércio eletrônico se tornou a forma predominante de comércio em todo o mundo. No Brasil, a Associação Brasileira de Comércio Eletrônico (ABCOMM) registrou um faturamento aproximado de 106 bilhões em 2020, com um crescimento esperado de 18% [ABCOMM 2020]. Para 2023, esse número se aproxima de R$160 bilhões. Entretanto, essa consolidação veio acompanhada por um aumento drástico no número de fraudes e golpes aplicados tanto nos clientes quanto nas empresas, o que torna a detecção de fraudes um fator essencial. Em linhas gerais, a fraude é um crime cujo objetivo é se apropriar do dinheiro de outros de forma ilícita, ou seja, ação onde um usuário (fraudador) obtém enriquecimento pessoal através do uso deliberado e indevido dos recursos de empresas e organizações.

Diante deste cenário, identificar, prever e detectar fraudes tornou-se uma tarefa essencial para empresas que atuam no comércio eletrônico. Para tanto, informações como: (i) o tempo gasto pelo usuário navegando no site; (ii) a frequência com que o usuário realize compras; (iii) o dispositivo frequentemente utilizado nas compras; (iv) o endereço IP padrão usado nos acessos; (v)as lojas mais visitadas; (vi) suas preferências de compra; (vii) outras informações como idade, sexo e localização (correlação de perfis), podem ser avaliadas para distinguir um usuário "normal" de um fraudulento. Em linhas gerais, é assim que as técnicas de detecção de fraude examinam conjuntos de dados para um

determinado site de comércio eletrônico e classificam um usuário como fraudulento ou não.

Apesar das várias soluções já estabelecidas para combater fraudes, a sofisticação das fraudes aumenta paralelamente ao aumento da quantidade de usuários em novas formas de comércio virtual. Os desafios que surgem mediante os métodos para detecção de fraude podem ser focados na extração de dados e a sua rotulação, e aspectos que não são bem estudados na literatura. Neste contexto, as redes *Graph Neural Networks* (GNN), métodos baseados em aprendizagem profunda, que operam no domínio gráfico, vêm ganhando destaque na literatura sobre detecção de fraude por permitirem, através da geração de comunidades de usuários com características semelhantes, identificar comportamentos fora do "normal".

O objetivo desta pesquisa é avaliar diferentes tipos de GNN, utilizando *datasets* de situações fraudulentas, a fim de montar e analisar redes de relacionamentos dos clientes e assim detectar possíveis fraudes em sistemas de comércio eletrônico. É importante destacar que o uso de GNN na possível detecção de fraudes se baseia nas capacidades deste tipo de rede neural de tanto identificar a representação de um nó central como em agrupar dados em comunidades para identificá-los, por exemplo, das preferências dos usuários ou outras características.

Para alcançar este objetivo, pretende-se: (i) criar ou adaptar uma base de dados reais para uso em futuras pesquisas, visto que hoje não existe nenhuma base deste tipo disponível na literatura acadêmica; (ii) definir uma ferramenta tecnológica para uso e experimentação envolvendo GNN; e (iii) conhecer e experimentar alguns algoritmos implementados GNN para fraude e seu comportamento em *databases* fraudulentas.

A metodologia utilizada para a realização desta pesquisa (Figura 1) foi composta por três fases: concepção, implementação e avaliação.

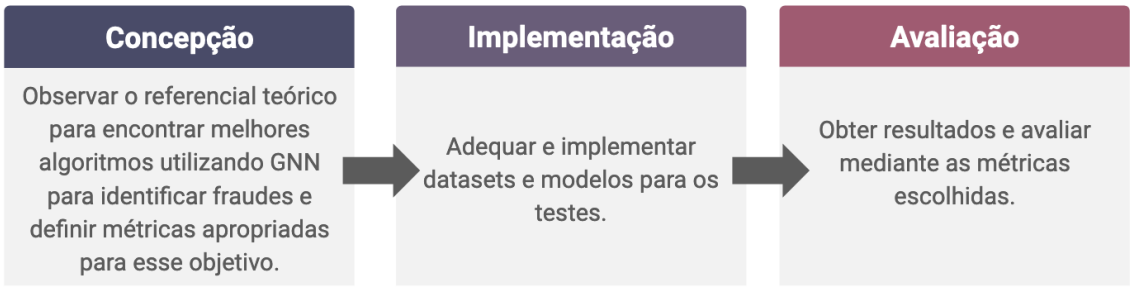

Figura 1. Metodologia da Pesquisa.

A concepção teve como foco utilizar o estado-da-arte para encontrar melhores algoritmos utilizando GNN. Para tanto, foi realizada uma revisão na literatura sobre GNN e algoritmos utilizados em detecção de fraudes, a fim de permitir escolher quais algoritmos de GNN são mais relevantes para esse trabalho. A implementação, o cerne do trabalho, consistiu na geração e/ou adequação de *dataset* para os algoritmos de GNN selecionados. Finalmente, a avaliação executou a experimentação, onde os algoritmos foram testados e os resultados obtidos, tomando como critérios de avaliação as métricas já estabelecidas na literatura.

2. Graph Neural Networks (GNNs)

Graph Neural Networks (GNNs), modelo proposto por [Scarselli et al. 2009], é uma extensão das redes neurais existentes para processar dados apresentados em forma de grafos. O objetivo do modelo é aprender o estado que todas as informações da vizinhança possuem para cada nó. Cada nó possui um vetor de características, que dependem da vizinhança e representa os conceitos ou objetos descritos por esse vetor. Os vértices são as relações entre os nós.

O processo de aprendizagem é feito através da interações entre os objetos e seus pesos, que se ajustam através do compartilhamento dos seus parâmetros. Essa interação gera um resultado que pode ser um rótulo de nó ou do grafo, dependendo da tarefa que foi designada, como classificação, predição de links, regressão, entre outros.

[Zeng and Tang 2021] detalham o funcionamento de uma GNN. Primeiro, a GNN seleciona nós vizinhos. Em seguida, uma função agregadora é aplicada para extrair informações ao redor do nó central. Por fim, a informação que foi agregada pela função é processada em uma rede neural, utilizando uma transformação não linear. A saída é uma representação atualizada do nó analisado. A Figura 2 ilustra esse processo.

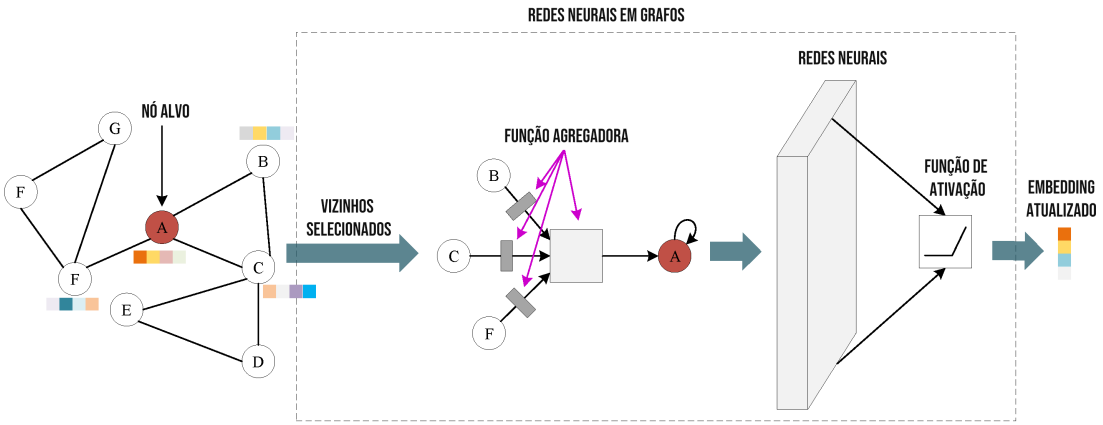

Figura 2. Processamento de uma GNN. Fonte: [Zeng and Tang 2021].

2.1. Tipos de GNN

A literatura descreve diferentes tipos de GNN voltadas para fraudes, com destaque para as GCN e GAT. As *Graph Convolutional Networks* (GCNs) foram propostas por [Kipf and Welling 2017] e generalizam as redes neurais convolucionais (CNNs) para dados representados em forma de grafo. Da mesma forma que as CNNs, as GCNs também executam uma operação linear (convolução) que envolve a multiplicação de um conjunto de pesos com os neurônios de entrada. Contudo, diferente das CNNs que apenas operam em dados de estruturas regulares (euclidiano), as GCNs conseguem atuar no âmbito não euclidiano. A *Graph Attention Networks* (GAT) foi proposta por [Veličković et al. 2018] e incorpora um mecanismo de atenção na etapa de propagação, que atribui pesos diferentes para os nós vizinhos, de modo que eles possam amenizar ruídos e obter melhores resultados.

2.2. Uso de GNN na detecção de fraudes

Na literatura acadêmica envolvendo fraudes, as GNNs podem ser utilizadas em duas funções: classificação de nós e detecção de comunidades. Na classificação de nós, cada nó é associado a um rótulo e, dado um grafo G parcialmente rotulado, o objetivo é utilizar esses nós rotulados para prever os rótulos dos não rotulados. Cada nó é representado como um vetor que contém as informações de sua vizinhança. Utilizando as características da vizinhança obtém-se uma classificação para os nós não classificados [Zhang et al. 2019](XU et al. 2019; WHU et al. 2019)

A outra aplicação é na detecção de comunidades. Uma vez que GNN permite a classificação de vértices, é possível fazer a detecção de comunidade. A ideia é que ao se definir uma comunidade como um grupo de vértices com conexões mais densas entre si, com relação a outras conexões com outras partes de um grafo G, é possível utilizar a rotulagem já obtida pela classificação de nós para separar os grupos de vertíces em comunidades.

3. Trabalhos Relacionados

Esta seção discute os trabalhos relacionados que criaram modelos de GNN especificamente para detecção de fraudes. [Zhang et al. 2019] desenvolveram o modelo **Player2Vec**, baseado em GCN, para detecção de crimes cibernéticos. A ideia foi codificar informações em cada relação, agregando vizinhos de diferentes relações. O modelo avalia o relacionamento entre usuários e identificar o principal ator nos grupos de fraudes, utilizando várias visualizações dos grafos - caracterizando as relações entre as personas.

[Li et al. 2019] utilizaram uma GCN para elaborar o modelo **GAS** (*GCN-based Anti-Spam method*) a fim de identificar SPAM em avaliações de produtos em páginas web. No modelo, os autores utilizam um grafo bipartido, composto por um grafo heterogêneo e um grafo homogêneo integrados, para capturar o contexto local e global de um comentário. [Wang et al. 2019] desenvolveram o modelo **FdGars** para identificar SPAM em avaliações utilizando um GCN semi-supervisionado de duas camadas, que combina características de texto, comportamento e relacionamento.

[LIU et al. 2018] apresentaram o modelo **GEM**, uma GNN Heterogênea, para detectar fraudes financeiras. A abordagem do modelo é inspirado em subgrafo conectado, que aprende de forma adaptativa com base em duas características dos suspeitos: agregação de dispositivos e agregação de atividades. Como esse tipo de grafo possui vários tipos de nós, também foi utilizado um mecanismo de atenção para aprender a importância entre diferentes nós. [Hu et al. 2019] desenvolveram o modelo **HACUD** (*Hierarchical Attention mechanism based Cash-out User Detection model*) que apresenta melhorias na representação das características dos objetos, explorando diferentes aspectos da rede de informações.

Uma vez que os modelos apresentados nesta seção são recentes e que o trabalho de [Liu et al. 2020] implementa, em uma *toolbox* chamada DGFraud, todos os modelos descritos para detecção de fraude baseados em GNN, optou-se por usar estes modelos nesta pesquisa. A Tabela 1 lista todos os modelos a serem avaliados.

4. Protocolo Experimental

Esta seção apresenta o ambiente, as métricas e os *datasets* empregados na pesquisa.

Tabela 1. Modelos Avaliados

Modelo	Tipo de Fraude	Grafo	Tipo
Player2Vec	Crimes Cibernéticos	Heterogêneo	GAT, GCN
GAS	Fraude em Opinião	Homogêneo	GCN, GAT
FdGars	Fraude Financeira	Homogêneo	GCN
GEM	Fraude Financeira	Heterogêneo	GCN
HACUD	Fraude Financeira	Heterogêneo	GAT

4.1. Ambiente

Todos os experimentos foram executados em Python 3.7.6, em um computador com Sistema Operacional Windows 10 Home, Core i5-10210U, 8GB ram, Nvidia Geforce MX250. Quanto ao software e bibliotecas, foi utilizado a plataforma de codigo aberto para *machine learning*, TensorFlow 1.15 e o ADAM, um otimizador de substituição para descida gradiente estocástica para modelos de *machine learning*, além da DGFraud.

Os modelos usaram taxa de aprendizagem em 0,001, momentum de 0,9 e 30 épocas para tempo de treinamento. Cada conjunto de dados é dividido em duas partes: 40% como o conjunto de treinamento e 60% como o conjunto de testes para avaliação de testes. Mais detalhes para os modelos estão no github do DGFraud (https://github.com/safe-graph/DGFraud).

4.2. Métricas de avaliação

Para o problema de fraude, a maior preocupação é a capacidade dos modelos em identificar corretamente os suspeitos, evitando erros e com previsibilidade correta. Por todos esses motivos, as métricas utilizadas neste trabalho são acurácia, F1-Score e AUC.

Acurácia mede quantas observações, positivas e negativas, foram classificadas corretamente. A fórmula da acurácia agrega tanto os falsos como os verdadeiros positivos e negativos. **F1 score** é uma métrica que combina precisão (corretude do modelo) e revocação (sensibilidade do modelo), calculando a média harmônica entre os dois. **AUC** (*Area under the ROC Curve*) pode ser interpretada como a probabilidade de que o modelo classifique um exemplo positivo aleatório mais alto do que um exemplo negativo aleatório. Maiores detalhes sobre essas métricas, especialmente em dados desbalanceados, podem ser encontrados em [Wardhani et al. 2019].

4.3. Datasets

Um *dataset* real foi obtido com uma empresa de e-commerce especializada em *cashbacking*. O *dataset*, totalmente anonimizado, contém as 27 características, incluindo identificador, sexo, data de nascimento, data de registro, indicações (identificadores de outros usuários indicados), movimentação monetária e um status de confiabilidade (fraudulento ou não). Em linhas gerais, ele contém 1.026.302 nós (usuários) e 45.606 arestas (indicações). Além disso, apenas 0.04% dos nós estão rotulados como fraudulentos, quase 97% como não fraudulentos e 3% como suspeitos e sem informação.

Para aplicar modelos GNN na avaliação deste *dataset*, há a necessidade de gerar listas adjacentes para representação que será usada nos algoritmos. A matriz adjacente gerada para o *dataset* foi de 413.925x413.925 com tamanho 1276 GB, o que impossibilitou

o processamento. É preciso esclarecer que esse tamanho da matriz se deve a quantidade de ligações entre os nós e a quantidade de características. Em outras palavras, quanto maior a base maior o processamento necessário para obter os resultados. Outro problema notado é que o *dataset* possui um grande desequilíbrio entre as classes para a classificação.

Devido a esses problemas, decidiu-se utilizar outro *dataset* para avaliar os modelos definidos para análise neste trabalho. O DBLP (*Digital Bibliography & Library Project*) é um *dataset* real que fornece uma lista abrangente de artigos de pesquisa em ciência da computação, onde os autores e co-autores são conectados entre si, tornando um *dataset* com grande gama de comunidades. Os dados de citação são extraídos do IEEE, ACM, MAG (*Microsoft Academic Graph*), e outros. Ele contém 629.814 artigos e 632.752 citações. Cada artigo tem o resumo, autores, ano, local e título. O grafo desse *dataset* possui 317.080 nós, com 1.049.866 conexões e 13.477 comunidades. As matrizes nesse dataset separadas em usuários (4057x4057), labels (4057x4) e features (4057x334). Vale destacar que embora não seja um *dataset* de fraudes, o DBLP permite testar os dois usos de GNN para fraudes (classificação de nós e detecção de comunidades). A principal relação entre os dois bancos de dados é que, no *cashback*, caso estudado aqui, sob a visão da empresa que ofereceu o banco de dados original, os suspeitos à usuários fraudulentos criavam comunidades utilizando-se de novas contas falsas conectadas à original para trazer retorno financeiro maior. Já o DBLP é um *dataset* que possui grandes agrupamentos em forma de comunidades, ainda que não possua o aspecto fraudulento.

5. Resultados

A Tabela 2 apresenta o resultado da avaliação dos modelos para a base DBLP, em relação às métricas F1-score, AUC e acurácia.

Tabela 2. Resultados da avaliação no DBLP.

Modelo	F1-Score	AUC	Acurácia	Tempo(s)
FdGars	0.7319	0.822	0.7319	19.622
GAS	0.5	0.5	0.5	0.594
GEM	0.295	0.5	0.75	6.923
Player2Vec	0.669	0.749	0.669	18.832
HACUD	0.7230	0.8576	0.7412	38.7547

Como pode ser observado na Tabela 2, os modelos FdGars e HACUD obtiveram os melhores desempenhos nas métricas, sendo o FdGars melhor na F1-Score e o HACUD melhor em AUC e acurácia. Infelizmente, o tempo de treinamento não é o forte desses modelos. O terceiro melhor modelo foi Player2Vec, com F1-score de 66%, 75% de AUC e quase 69% de acurácia. Por fim, os modelos GAS e GEM apresentaram os piores resultados, com o GAS obtendo 50% em todas as métricas e o GEM obtendo 29%, 50% e 75% em F1-Score, AUC e acurácia, respectivamente.

Discutindo melhor os resulados, o FdGars, que faz uma classificação a prévia de similaridade entre os conteúdos, comportamento dos usuários e relacionamento entre os usuários mostrou-se uma abordagem apropriada no contexto da DBLP (existe similaridade em resumos e rótulos de autores na mesma área, e ligações entre conferências, artigos e autores). Outra característica do FdGars é possuir auto-classificação de um nó

fraudulento através dos padrões identificados no grafo, característica observada no DBLP e que pode ser analisada no *dataset* de e-commerce.

O modelo HACUD tem uma abordagem diferente que utiliza um mecanismo de atenção hierárquico para observar características de diferentes importâncias. Como o DBLP apresenta ligações entre conferências, artigos e autores, gerando uma relação complexa ideal para o mecanismo de atenção hierárquica.

É importante destacar que tanto o modelo FdGars quanto o HACUD propõem formas de identificação de um nó fraudulento. Ambos são aplicáveis nas características do *dataset* de e-commerce, onde existe a conexão entre usuários para gerar interação e o comportamento de nós suspeitos se estende aos nós associados. Sobre o tempo de execução de ambos, pode-se assmuir que a forma de operação do modelo HACUD acarrete esse tempo maior. Já para o FdGars e Player2Vec, suas abordagens explica o tempo elevado.

Por fim, o resultado do modelo GAS se justifica pela operação que agrega diretamente vizinhos com diferentes tipos de nós. Essa abordagem claramente não é adequada ao escopo de fraudes. Já o modelo GEM, testado em *dataset* real de fraudes, deixa a dúvida se o modelo não estava em *overfitting* em relação ao *dataset* do artigo e, por isso, precisa de melhores ajustes (*tuning*) para generalizar seus resultados eno DBLP.

6. Conclusão

Este trabalho estudou vários modelos de GNN para detecção de fraudes em e-commerce, definidos através de um levantamento bibliográfico sobre os modelos atuais nesta área e detalhamento dos seus mecanismos para esse tipo de classificação.

Foram avaliados cinco (5) modelos Player2Vec, GAS, FdGars, GEM e HACUD, utilizando o *dataset* DBLP. Os modelo foram mensurados sob acurácia, F1-score, AUC e tempo de execuçãom, onde a AUC avaliou os problemas com distribuição de amostras muito disparates, já que é uma medida invariante, e o F1-score permitiu uma visualização da sensibilidade e corretude do modelo escolhido. Todas as escolhas foram focadas no *tradeoff* entre a experiência do usuário e eficiência em eliminar contas maliciosas para que não haja perdas financeiras. Os resultados indicaram que todos os modelos têm F1-score entre 30% a 73% e AUC de 88% a 50%, com HACUD, FdGars e Player2Vec com melhores AUC e F1-score, o que pode indicar que utilizar mecanismos de atenção é um boa abordagem para selecionar e distinguir diferentes relações entre nós, e tomar ações mais eficazes na detecção de fraudes. Sobre os tempos dos modelos, há necessidade de otimizações para processamento de grandes grafos originados de *datasets* reais.

Falando sobre contribuições e trabalhos futuros, hoje existe uma base com dados reais nacionais de e-commerce, que ainda precisa de ajuste para ser liberada. Outro ponto interessante foi a identificação de ferramental simples e gratuito para uso e experimentação envolvendo GNN (DGFraud). Por fim, o principal trabalho futuro envolvendo essa pesquisa é a análise do *dataset* de e-commerce, visto que uma vez que o problema de geração seja resolvido, acredita-se que mais modelos poderão ser analisados.

7. Agradecimentos

Esta pesquisa foi realizada com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Referências

ABCOMM (2020). O comércio eletrônico deve crescer 18% em 2020 e movimentar r$ 106 bilhões. https://bityli.com/3DjEH.

Hu, B., Zhang, Z., Shi, C., Zhou, J., Li, X., and Qi, Y. (2019). Cash-out user detection based on attributed heterogeneous information network with a hierarchical attention mechanism. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):946–953.

Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26*. OpenReview.net.

Li, A., Qin, Z., Liu, R., Yang, Y., and Li, D. (2019). Spam review detection with graph convolutional networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. ACM.

LIU, Z., CHEN, C., YANG, X., ZHOU, Junand LI, X., and SONG, L. (2018). Heterogeneous graph neural networks for malicious account detection. In *27th ACM International Conference On Information And Knowledge Management*, pages 2077–2085.

Liu, Z., Dou, Y., Yu, P. S., Deng, Y., and Peng, H. (2020). Alleviating the inconsistency problem of applying graph neural network to fraud detection. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1569–1572, New York, NY, USA. ACM.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2018). Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Conference Track Proceedings*. OpenReview.net.

Wang, J., Wen, R., Wu, C., Huang, Y., and Xion, J. (2019). Fdgars: Fraudster detection via graph convolutional networks in online app review system. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, page 310–316, New York, NY, USA. ACM.

Wardhani, N. W. S., Rochayani, M. Y., Iriany, A., Sulistyono, A. D., and Lestantyo, P. (2019). Cross-validation metrics for evaluating classification performance on imbalanced data. In *2019 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, pages 14–18.

Zeng, Y. and Tang, J. (2021). Rlc-gnn: An improved deep architecture for spatial-based graph neural network with application to fraud detection. *Applied Sciences*, 11(12).

Zhang, Y., Fan, Y., Ye, Y., Zhao, L., and Shi, C. (2019). Key player identification in underground forums over attributed heterogeneous information network embedding framework. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 549–558, New York, NY, USA. ACM.