Санкт–Петербургский государственный университет

ПИВНЕВ Игорь Алексеевич

Отчет о научно-исследовательской работе

Использование нейросетевых методов в прикладных задачах определения мошенничества

Уровень образования: бакалавриат
Направление 01.03.02 «Прикладная математика и информатика»
Основная образовательная программа СВ.5005.2015 «Прикладная математика, фундаментальная информатика и программирование»

Научный руководитель:
Кандидат физико-математических наук,
доцент кафедры теории систем управления
электрофизической аппаратурой
ГОЛОВКИНА Анна Геннадьевна

Санкт-Петербург
2023 г.

Содержание

Введение

С каждым днем цифровое пространство становится все более неотъемлемой частью нашей повседневной жизни. В современной экономике растет влияние мошенничества, особенно в контексте быстрого распространения социальных сетей и развития финансовых технологий в форме необанкинга. Этот развитие акцентирует внимание на нехватке эффективных методов борьбы с киберугрозами. Исследование нейросетевых методов обнаружения аномалий и выявления паттернов мошеннической активности становится не только актуальным, но и стратегически важным для обеспечения цифровой безопасности.

Выбор нейронных сетей обусловлен уникальными возможностями повышения точности систем безопасности, адаптации к сложным и динамичным структурам данных, что делает их эффективным средством анализа в условиях постоянно изменяющейся природы цифрового мошенничества. Применение таких подходов может оказать существенное влияние на различные сферы:

- Электронная коммерция: улучшение алгоритмов обнаружения мошенничества в банковских финансовых операциях и онлайн-торговле, повышение скорости нахождения схем и махинаций с платежами.
- Кибербезопасность: разработка инновационных методов выявления аномалий в сетевом трафике и предотвращения цифровой преступности с фокусом на оперативное реагирование на новые угрозы и сценарии атак.
- Онлайн-медиа: использование нейросетей для выявления фейковых новостей и манипуляций с медийными данными, улучшения общей достоверности информации, предоставляемой онлайн-ресурсами.

Постановка задачи

В данной работе будем считать мошенническим любое действие, которое влечет за собой кражу финансовых средств, системной или личной информации, а также любое другое нарушение закона или общественного правопорядка.

Основной целью является изучение применения различных архитектур нейронных сетей в прикладных задачах определения подозрительных активностей. Для достижения этого ставятся следующие подзадачи:

1. Анализ подходов: исследовать существующие нейросетевые методы и выделить их ключевые особенности.

2. Описание данных: рассмотреть различные способы представления информации для поиска наиболее релевантных признаков приминительно к каждому конкретному подходу.

3. Измерение эффективности: изучить основные метрики и выбрать наиболее подходящие для оценки результатов.

Глава 1. Графовые свертки

1.1 Пространственная парадигма

В актуальных исследованиях по определению мошенничества применяются различные методы, такие как графовые свертки [1]. вероятностные модели и анализ журнала изменений (log-file) [11] с использованием больших языковых моделей (Large Language Model, LLM) [10]. Классические подходы машинного обучения оказываются ограниченными в эффективности при обработке сложных мошеннических сценариев, где последние могут изменять свои тактики или объединяться в организованные группировки.

Графовая сверточная нейронная сеть (Graph Convolutional Neural Network, GCNN) — это обобщение широкоизвестных сверточных нейросетей. GCNN способны учитывать взаимосвязи между узлами, что является важным аспектом при глубоком анализе сетей мошенничества, однако может стать ограничением при обработке больших данных.

Граф принято представлять как пару множеств $G(V, E)$, где первое содержит все вершины и их признаковые описания, часто также именуемые эмбеддингами (embeddings) или скрытыми представлениями, а второе все ребра и характеристики связей. Окрестность вершины v обычно обозначается как $N(v)$. Граф может быть ориентированный или ненаправленный, полносвязный или многодольный, с одним отношением между узлами или множеством, а также гомо- или гетерогенный. Тогда свертка описывается следующим алгоритмом [1]:

1. Для каждой вершины находится набор скрытых представлений соседних вершин $h_{N(v)}$, из которых идут связи в текущую.

2. Собранная информация агрегируется с помощью коммутативной операции $AGGR$ (например, взятие средних или максимальных значений скрытых представлений объектов) в вектор фиксированного размера.

3. Полученный вектор объединяется со скрытым представлением вершины h_v и они домножаются на обучаемую матрицу W.

4. К результату в качестве нелинейной операции поэлементно применяется сигмоидальная функця:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Компактно алгоритм можно записать следующими двумя формулами:

$$m_v^{t+1} = \text{AGGR}\left(\{h_w^t, w \in N(v)\}\right)$$

$$h_v^{t+1} = \sigma\left(W^{t+1}\,\text{CONCAT}(m_v^{t+1}, h_v^t)\right)$$

Описанный выше подход обычно называется пространственной парадигмой. В нем обучаемые параметры содержат только одну матрицу W. Самой известной реализацией этого подхода, предложенной еще до распространения нейросетевых методов, был алгоритм PageRank, который был предложен и активно использовался компанией Google для ранжирования поисковой выдачи.

Такая свертка использует только скрытые представления вершин, однако уделяет больше внимания локальному окружению, нежели глобальному положению вершины во всем графе. Авторы многих работ [1][4][5][6][7] доказали высокое качество данной архитектуры в задачах, связанных с выучиванием представлений вершин, однако использование данной свертки можно встретить и в других задачах, связанных с обработкой графов, не содержащих дополнительной информации о рёбрах.

С развитием архитектуры трансформеров и концепции внимания (attention) [14][15] появились и модификации алгоритма свертки [2][3], которые в общем виде можно описать следующими двумя формулами:

$$\alpha_v = \text{softmax}\left(\text{act}\left(\mathbf{a}^T\text{CONCAT}(W_{hv}^T, W_{h^*t})\right)\right)$$

$$h_{v,t+1} = \sigma\left(\sum_{w \in N(v)} \alpha_{vw} W_{hw}^t\right)$$

Где act -– некоторая функция активации, например широкоприменимая ReLU. В момент обновления представления вершины attention генерирует веса α_{vw}, которые указывают, информация из каких вершин важнее. Формула $softmax$ позволяет перевести вещественные ответы нейросети в вероятности:

$$\text{softmax}(x) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

1.2 Спектральная парадигма

В качестве альтернативы пространственной парадигме выступает спектральная, которая основана на анализе процесса диффузии сигнала внутри графа при помощи матрицы смежности и лапласиана графа.

Лапласиан графа — это матрица $L = D - A$, где диагональная D хранит в i-й ячейке количество исходящих из i-й вершины рёбер и A — матрица смежности графа, a_{ij} элемент которой равен числу рёбер, соединяющих i-ю и j-ю вершину.

Лапласиан имеет неотрицательные собственные значения, среди которых количество нулевых всегда совпадает с количеством компонент связности графа, а собственные векторы, соответствующие положительным собственным значениям, описывают разрезы графа – его разделения пополам так, чтобы между разделенным половинами было как можно меньше рёбер.

Этим свойством Лапласиана графа пользуются для того, чтобы проводить кластеризацию графа без учителя. Для этого надо:

1. Посчитать Лапласиан L матрицы A.
2. Найти k собственных векторов, соответствующих наименьшим собственным значениям.
3. Сформировать матрицу размера $n \times k$.
4. Кластеризовать объекты, описываемые этой матрицей (например, с помощью алгоритма K-Means).

Таким образом, спектральный подход отлично подходит в задачах, связанных с обработкой одного большого графа, где важно понимать относительное месторасположение вершины в этом большом графе.

Глава 2. Другие подходы

Трансформеры [10][14][15], основанные на механизме attention, успешно применяются для моделирования последовательностей и выявления зависимостей в журналах событий. Лог-файлы представляют собой зарегистрированые статусы работы программы, которые обычно служат для проверки или отладки системы.

Изначально трансформеры были спроектированы для решения задач обработки естественного языка (Natural Language Processing, NLP). Однако это позволило достичь хороших результатов при работе с большими объемами данных и выявлять аномалии, свидетельствующие о мошеннической активности, но такой подход требует значительных вычислительных ресурсов.

Наконец, самым простым и распространенным методом в задаче определения мошенничества является использование статистик, однако такой подход не поддается общему описанию и зависит от конкретно определенной сферы применения.

В качестве примера набора данных и формулировки конкретной задачи, можно рассмотреть популярные Amazon-Fraud и Yelp-Fraud [7]. Они собирались из широкоизвестных интернет-ресурсов и часто служат основным способом оценки работы цитируемых архитектур [3][4][6].

Amazon-Fraud содержит информацию об отзывах, оставленных на музыкальные инструменты посетителями сайта. Узлы представляют собой пользователей, которые соединены по трем видам отношений: отзыв на общий товар, одинаковый рейтинг или содержание описания (для сравнения использовали TF-IDF).

Yelp-Fraud структурно обратен – здесь тоже граф, но в узлах его находятся отзывы на рестораны и отели, а ребра используются, если: отзывы было от одного пользователя, в одно время или с одинаковым рейтингом.

Оба набора данных используются в задачах определения мошенничества, так как под недобросовестной деятельностью в таком случае можно считать спам или намеренное занижение оценки товара.

Заключение

В данной работе рассмотрены различные подходы, используемые в прикладных задачах определения мошенничества. На основании проведенного исследования можно сделать вывод, что применение машинного обучения обладает потенциалом для повышения точности и скорости обнаружения недобросовестной деятельности.

Выделяются перспективы интеграции нейросетевых методов в существующие системы противодействия цифровой преступности. Однако, для успешной реализации таких подходов, необходимо учитывать особенности данных, обеспечивать их надежное хранение и адаптивность системы к новым схемам и видам мошенничества.

Следующим шагом будет воспроизведение результатов, которых достигли цитируемые авторы в своих работах, а также тестирование архитектур на едином наборе данных из финансовой сферы для последующего сравнительного анализа. Затем планируется переприменить наилучшие из рассмотренных подходов к новой задаче, основанной на предварительно обработанных и обезличенных данных, предоставленных моим работодателем. Последние взяты из приложения социальной-сети, в которой мошенником будет считаться спам-робот.

Список литературы

[1] Kipf T. N., Welling M. «Semi-Supervised Classification with Graph Convolutional Networks». 2016, DOI: https://arxiv.org/abs/1609.02907.

[2] Cheng D., Ye Y., Xiang S., Ma Z., Zhang Y., Jiang C. «Anti-Money Laundering by Group-Aware Deep Graph Learning». IEEE, 2023, pp. 12444-12457. DOI: https://ieeexplore.ieee.org/document/10114503.

[3] Wang Y., Zhang J., Huang Z., Li W., Feng S., Ma Z., Sun Y., Yu D., Dong F., Jin J., Wang B., Luo J. «Label Information Enhanced Fraud Detection against Low Homophily in Graphs». ACM, 2023. DOI: https://dx.doi.org/10.1145/3543507.3583373.

[4] Xiang S., Zhu M., Cheng D., Li E., Zhao R., Ouyang Y., Chen L., Zheng Y. «Semi-Supervised Credit Card Fraud Detection via Attribute-Driven Graph Representation». AAAI, 2023, DOI: https://doi.org/10.1609/aaai.v37i12.26702.

[5] Hu S., Zhang Z., Luo B., Lu S., He B., Liu L. «BERT4ETH: A Pre-trained Transformer for Ethereum Fraud Detection». ACM, 2023. DOI: https://dx.doi.org/10.1145/3543507.3583345.

[6] Qin Z., Liu Y., He Q., Ao X. «Explainable Graph-Based Fraud Detection via Neural Meta-Graph Search». CIKM, 2022, pp. 4414-4418, DOI: https://doi.org/10.1145/3511808.3557598.

[7] Dou Y., Liu Z., Sun L., Deng Y., Peng H., Yu P.S. «Enhancing Graph Neural Network-based Fraud Detectors against Camouflaged Fraudsters». ACM, 2020. DOI: https://dx.doi.org/10.1145/3340531.3411903.

[8] Kireev K., Andriushchenko M., Troncoso C., Flammarion N. «Transferable Adversarial Robustness for Categorical Data via Universal Robust Embeddings». 2023, DOI: https://arxiv.org/abs/2306.04064.

[9] Shafahi A., Huang W.R., Studer C., Feizi S., Goldstein T. «Are adversarial examples inevitable?». 2020, DOI: https://arxiv.org/abs/1809.02104.

[10] Pan J., Wong S.L., Yuan Y. «RAGLog: Log Anomaly Detection using Retrieval Augmented Generation». 2023, DOI: https://arxiv.org/abs/2311.05261.

[11] Zhu J., He S., He P., Liu J., Lyu M. R. «Loghub: A Large Collection of System Log Datasets for AI-driven Log Analytics». 2020, DOI: https://arxiv.org/abs/2008.06448.

[12] Jesus S., Pombal J., Alves D., Cruz A., Saleiro P., Ribeiro R.P., Gama J., Bizarro P. «Turning the Tables: Biased, Imbalanced, Dynamic Tabular Datasets for ML Evaluation». 2022, DOI: https://arxiv.org/abs/2211.13358.

[13] Wu B., Yao X., Zhang B., Chao K.-M., Li Y. «SplitGNN: Spectral Graph Neural Network for Fraud Detection against Heterophily». CIKM, 2023, pp. 2737-2746, DOI: https://doi.org/10.1145/3583780.3615067.

[14] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser L., Polosukhin I. «Attention Is All You Need». arXiv, 2017, DOI: https://arxiv.org/abs/1706.03762.

[15] Devlin J., Chang M.-W., Lee K., Toutanova K. «BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding». 2018, DOI: https://arxiv.org/abs/1810.04805.