1 Міжнародна науково-практична конференція «Modern Perspectives on Science and Economic Progress»

Інформаційні технології та кібербезпека

МЕТОДИ РЕАЛІЗАЦІЇ ДВОЕТАПНОЇ CNN-АРХІТЕКТУРИ ДЛЯ КЛАСИФІКАЦІЇ АТАК У МЕРЕЖАХ IOT

Рихва Володимир,
Аспірант
volodymyr.rykhva@hneu.net
Солодовник Ганна
к.т.н., доцент
ganna.solodovnyk@hneu.net
кафедри кібербезпеки та інформаційних технологій
Харківський національний економічний
університет імені Семена Кузнеця, Україна

Стрімке зростання кількості пристроїв Інтернету речей (IoT) характеризується експоненціальним зростанням кількості підключених пристроїв. Ця тенденція супроводжується диверсифікацією кіберзагроз, включаючи DDoS-атаки, ботнети, APT та атаки на протоколи IoT [1]. Особливістю IoT-середовищ є гетерогенність пристроїв, обмежені обчислювальні ресурси та різноманітність мережевих протоколів, що створює унікальні виклики для систем виявлення вторгнень [2].

Конволюційні нейронні мережі (CNN) зарекомендували себе як ефективний інструмент для аналізу мережевого трафіку завдяки здатності автоматично виявляти просторові патерни. Проте традиційні одноетапні CNN-архітектури стикаються з компромісом між точністю та швидкістю обробки: моделі з високою точністю часто мають значні затримки, що неприйнятно для застосувань у реальному часі в IoT мережах [4].

Двоетапні архітектури пропонують потенційне вирішення цієї проблеми шляхом адаптивного розподілу обчислювальних ресурсів залежно від складності класифікації зразка. Перший етап виконує швидку попередню класифікацію, тоді як другий етап здійснює детальний аналіз лише для зразків, що викликають сумніви. Цей підхід дозволяє досягти балансу між точністю та швидкістю обробки, що є критично важливим для ефективного виявлення атак у мережах IoT.

Варіанти реалізації двоетапних CNN-архітектур

1. Послідовна фільтрація (Sequential Filtering)

Цей підхід передбачає використання легкої моделі на першому етапі для швидкої фільтрації трафіку. Зразки, які викликають сумніви, передаються на другий етап, де застосовується більш потужна модель для детального аналізу [5]. Такий метод дозволяє ефективно розподіляти обчислювальні ресурси, знижуючи загальну латентність системи. Застосування цього підходу в системах виявлення вторгнень дозволяє досягти балансу між швидкістю обробки та точністю класифікації, що є критично важливим для ресурсно обмежених IoT-пристроїв.

2. Ієрархічна класифікація (Hierarchical Classification)

У цьому підході перший етап класифікує трафік за широкими категоріями атак, наприклад, DDoS, Malware. Другий етап спеціалізується на підтипах кожної категорії, забезпечуючи більш точну класифікацію [6]. Цей метод дозволяє краще враховувати доменні знання та структуру загроз. Такий підхід дозволяє зменшити ймовірність помилкової класифікації та покращити загальну продуктивність системи виявлення вторгнень.

3. Адаптивна глибина (Adaptive Depth)

Цей підхід використовує одну глибоку CNN з можливістю раннього виходу для простих випадків. Зразки, які потребують більш детального аналізу, проходять через всю глибину мережі [7]. Це дозволяє зменшити середню затримку системи без втрати точності для складних випадків. Застосування адаптивної глибини в системах виявлення вторгнень забезпечує ефективне використання обчислювальних ресурсів та зменшення часу обробки, що є важливим для реального часу в IoT-середовищах.

4. Ансамблева спеціалізація (Ensemble Specialization)

Метод ансамблевої спеціалізації передбачає побудову системи виявлення вторгнень, яка складається з кількох спеціалізованих класифікаторів, кожен з яких оптимізований для виявлення певного типу атак [8]. На першому етапі застосовується мета-класифікатор, який визначає загальну категорію загрози та спрямовує зразок до відповідного спеціалізованого класифікатора для детального аналізу. Такий підхід дозволяє досягти високої точності виявлення, оскільки кожен класифікатор налаштований на специфічні характеристики певного типу атак. Однак, варто зазначити, що ансамблева спеціалізація вимагає значних обчислювальних ресурсів для тренування та підтримки кількох моделей, а також для координації між мета-класифікатором та спеціалізованими класифікаторами.

5. Гібридні архітектури (Hybrid Architectures)

Гібридні архітектури поєднують різні типи нейронних мереж для покращення детекції атак.

5.1 CNN + LSTM

Гібридна архітектура, що поєднує CNN та LSTM, дозволяє ефективно виявляти як просторові, так і часові патерни в мережевому трафіку. CNN використовується для витягування локальних ознак, тоді як LSTM моделює часові залежності, що особливо корисно для виявлення атак, які розвиваються протягом часу [9]. Такий підхід забезпечує високу точність виявлення атак у мережах IoT, особливо для складних сценаріїв з часовими залежностями, таких як атаки типу DDoS або ботнети.

5.2 CNN + XGBoost

У цій гібридній архітектурі CNN використовується для витягування ознак з мережевого трафіку, після чого XGBoost застосовується для класифікації. CNN ефективно обробляє високорозмірні дані, зменшуючи їх до релевантних ознак, які потім передаються до XGBoost для точного виявлення атак [10]. Такий підхід поєднує переваги глибокого навчання та градієнтного бустингу, забезпечуючи високу точність виявлення атак при збереженні ефективності обчислень, що є важливим для IoT-середовищ з обмеженими ресурсами.

5.3 CNN + ViT (Vision Transformer)

Гібридна архітектура CNN + ViT (Vision Transformer) поєднує переваги згорткових нейронних мереж (CNN) та трансформерів для обробки зображень. У цьому підході CNN використовується для витягування локальних ознак з мережевого трафіку, тоді як ViT моделює глобальні залежності, що дозволяє ефективно виявляти як локальні, так і глобальні патерни атак [11]. Застосування такої гібридної архітектури забезпечує високу точність виявлення атак у мережах IoT, особливо для складних сценаріїв з часовими та просторовими залежностями.

Для порівняння розглянутих двоетапних CNN-архітектур (Табл.1) використано кілька основних критеріїв. Точність (F1-score) оцінює, наскільки добре модель класифікує різні типи атак, затримка (Inference Time) описує час обробки одного зразка, а складність архітектури враховує кількість компонентів та взаємодій. Потреба в обчислювальних ресурсах визначає вимоги до пам’яті та потужності, адаптивність відображає здатність підлаштовуватися під нові загрози, а стійкість до дисбалансу класів показує, як модель працює з рідкісними класами. Модульність характеризує легкість розширення чи оновлення архітектури.

Табл.1 – Порівняльна характеристика двоетапних CNN-архітектур

Критерій	Sequential Filtering	Hierarchical Classification	Adaptive Depth	Ensemble Specialization	CNN + ViT	CNN + LSTM	CNN + XGBoost
Точність	Середня–висока	Висока	Висока	Дуже висока	Дуже висока	Висока	Висока
Затримка	Низька	Середня	Дуже низька	Висока	Висока	Середня	Середня–висока

Складність архітектури	Низька–середня	Середня–висока	Середня	Дуже висока	Дуже висока	Середня	Середня
Потреба в ресурсах	Середня	Висока	Низька	Дуже висока	Дуже висока	Середня	Середня
Адаптивність до нових атак	Обмежена	Середня	Висока	Дуже висока	Висока	Середня	Висока
Стійкість до дисбалансу	Середня	Висока	Середня	Висока	Висока	Середня	Дуже висока
Інтерпретованість	Висока	Середня	Середня	Середня	Низька	Середня	Висока
Модульність	Обмежена	Середня	Висока	Дуже висока	Середня	Середня	Середня

Список використаних джерел

1. Ferrag, M. A., Friha, O., Hamouda, D., Maglaras, L., & Janicke, H. (2024). Securing internet of things using machine and deep learning methods: A survey. *Cluster Computing*, 27(3), 2651-2696. https://doi.org/10.1007/s10586-024-04509-0Zhong, M., Lin, M., Zhang, C., & Xu, Z. A survey on graph neural networks for intrusion detection systems: methods, trends and challenges. Computers & Security, 141, 103821, 2024.

2. Alotaibi, B., & Kumar, S. (2024). A novel intrusion detection framework for optimizing IoT security. *Scientific Reports*, 14, 21234. https://doi.org/10.1038/s41598-024-72049-z

3. Rahman, A., Islam, M. J., Band, S. S., Muhammad, G., Hasan, K. F., & Tiwari, P. (2024). Enhancing IoT network security through deep learning-powered intrusion detection system. *Digital Communications and Networks*, 10(4), 887-901. https://doi.org/10.1016/j.dcan.2023.08.009

4. Almalawi, A., Yu, X., Tari, Z., Fahad, A., & Khalil, I. (2024). An efficient CNN-based intrusion detection system for IoT: Use case towards cybersecurity. *Electronics*, 13(10), 1876. https://doi.org/10.3390/electronics13101876

5. Wang, L., & Zhang, Y. (2025). A two-tier optimization strategy for feature selection in robust adversarial attack mitigation on internet of things network security. Scientific Reports, 15, 3567. https://doi.org/10.1038/s41598-025-85878-3

6. Rodriguez-Garcia, M., Santos, I., Bringas, P. G., & Galar, M. (2024). A deep learning ensemble approach to detecting unknown network attacks. Journal of Network and Computer Applications, 198, 103287. https://doi.org/10.1016/j.jnca.2022.103287

7. Teerapittayanon, S., McDanel, B., & Kung, H. T. (2016). BranchyNet: Fast inference via early exiting from deep neural networks. 2016 23rd International Conference on Pattern Recognition (ICPR), 2464–2469. https://doi.org/10.1109/ICPR.2016.7897017

8. Sayem, I. M., Sayed, M. I., Saha, S., & Haque, A. (2024). ENIDS: A Deep Learning-Based Ensemble Framework for Network Intrusion Detection Systems. IEEE Transactions on Network and Service Management. https://doi.org/10.1109/TNSM.2024.3414305

9. Khan, L. U., Yaqoob, I., Imran, M., Han, Z., & Guizani, S. (2021). 6G wireless systems: A vision, architectural elements, and future directions. IEEE Access, 8, 147029–147044. https://doi.org/10.1109/ACCESS.2020.3015289

10. Xu, H., Zhang, Y., & Wang, H. (2022). Hybrid deep learning and XGBoost model for network intrusion detection. IEEE Access, 10, 57723–57733. https://doi.org/10.1109/ACCESS.2022.3179180

11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv:2010.11929. https://doi.org/10.48550/arXiv.2010.11929