DOI:**10.33099/2311-7249/2023-47-2-113-118**

УДК 519.682

Ільїн Дмитро Володимирович
Старинський Іван Михайлович *(кандидат технічних наук)*

Національний університет оборони України, Київ, Україна

МАТЕМАТИЧНА МОДЕЛЬ СИСТЕМИ ВИЯВЛЕННЯ ВТОРГНЕНЬ З ВИКОРИСТАННЯМ НЕЙРОННОЇ МЕРЕЖІ НА ОСНОВІ АВТОЕНКОДЕРІВ

Інформаційно-телекомунікаційна мережа військового призначення має великий обсяг наборів даних, а забезпечення захищеності такої мережі від кібератак, є працеємним процесом. Дані мережевого трафіку мають складні нелінійні зв'язки, що змінюються в часі. Існуючі моделі забезпечення кіберзахищеності базуються на моделях кореляції даних про трафік і вимагають значних обчислювальних витрат та не дають змоги здійснювати обробку мережевого трафіку в реальному часі. Крім того, вони не враховують просторово-часові кореляції даних. Метою статті є розроблення математичної моделі системи виявлення вторгнень на основі мережі автоенкодерів для забезпечення кіберзахищеності інформаційно-телекомунікаційної мережі військового призначення. Запропоновано розроблену математичну модель системи виявлення вторгнень на основі нейронної мережі, яка базується на поєднанні багатошарової згорткової нейронної мережі на основі автоенкодерів з використанням довгострокової короткочасної пам'яті. Розроблена модель системи виявлення вторгнень спочатку використовує багатошарову згорткову нейронну мережу на основі автоенкодерів для аналізу просторових особливостей набору даних, які потім обробляються автоенкодерами з використанням довгострокової короткочасної пам'яті для виявлення аномалій у мережевому трафіку. Для підвищення точності виявлення вторгнень запропоновано застосовувати два алгоритми Isolation Forest, що виправляють помилки, виявляють хибнопозитивні та хибнонегативні результати. Тренування моделі системи виявлення вторгнень на основі нейронної мережі проводилось з використанням набору даних NSL-KDD та показало високу точність реконструкції даних та її працездатність.

Ключові слова: *інформаційно-телекомунікаційна мережа, кіберзахищеність, нейронна мережа, система виявлення вторгнень, автоенкодер.*

Вступ

Постановка проблеми. За умов стрімкого зростання кіберризиків і кіберзагроз важливим є питання забезпечення кіберзахисту інформаційно-телекомунікаційних мереж (далі – ІТМ) військового призначення (далі – ВП). Особливої уваги потребують DoS-атаки, які є найбільш небезпечними, простими в організації та найдешевшими за вартістю кіберзагрозами. Так, наприклад, в Україні такі кібератаки здійснювалися на сайти органів державної влади, а саме, Президента України, Кабінету Міністрів України, Міністерства оборони України, Служби безпеки України, Міністерства внутрішніх справ України тощо. Водночас було встановлено більше 5 тисяч кібератак. Ці кібератаки показали низький рівень кіберзахищеності ІТМ від такого типу запланованих кібератак, та відсутність досить ефективних засобів захисту, таких як системи виявлення вторгнень [1]. У зв'язку з цим, розроблення методів і моделей, що дають змогу формалізувати процеси виявлення вторгнень є актуальним науковим завданням.

Аналіз останніх досліджень та публікацій. Аналіз літератури з кібербезпеки свідчить, що для забезпечення кіберзахищеності ІТМ створюються системи виявлення вторгнень (далі – СВВ) на основі нейронних мереж (далі – НМ). Наприклад, у роботах [2–4] пропонується СВВ нейронної мережі з неконтрольованим навчанням, що базується на напівконтрольованій нечіткій C-Mean кластеризації з одношаровими НМ прямого зв'язку (далі – ПЗ), також відомою як Extreme Learning Machine (далі – ELM) для виявлення вторгнень у режимі реального часу.

Сьогодні використовуються методи неконтрольованого глибинного навчання, такі як мережа глибоких переконань (Deep Belief Network) (далі – DBN), самоорганізуючі карти та автоенкодери. У роботі [5] запропоновано модель СВВ нейронної мережі на основі самоорганізованої карти, яка покращує виявлення вторгнень. А в роботах [6] та [7] запропоновано СВВ нейронної мережі з неконтрольованим глибинним навчанням DBN для виявлення вторгнень. Крім того, у значній кількості досліджень вивчалося застосування глибокої мережі переконань у проєктуванні СВВ НМ [8; 9].

Разом із тим, досвід побудови СВВ НМ свідчить, що перспективним напрямом дослідження є побудова ефективних СВВ НМ із використанням автоенкодерів (далі – АЕ), оскільки вони прості у реалізації та маловартісні. У низці

досліджень була спроба розробити варіанти АЕ з покращеними характеристиками щодо виявлення вторгнень. Разом із тим, стає очевидним, що, незважаючи на значний приріст продуктивності, досягнутий із застосуванням СВВ з неконтрольованим навчанням, має місце їх достаньо низька ефективність стосовно виявлення прихованих кібератак. Таким чином, дослідження проблеми забезпечення кіберзихищеності інформаційно-телекомунікаційних системи від кібератак на основі СВВ НМ з неконтрольованим навчанням на основі автоенкодера є актуальним.

Метою статті є розроблення математичної моделі системи виявлення вторгнень з використанням нейронної мережі на основі автоенкодерів для забезпечення кіберзахищенності інформаційно-телекомунікаційної мережі військового призначення.

Виклад основного матеріалу дослідження

Для забезпечення кіберзахищеності ІТМ доцільно використовувати СВВ на основі автоенкодерів, що є одним із типів НМ прямого зв'язку з неконтрольованим навчанням (без вчителя) та застосовується для реконструкції вхідних даних. Нейронна мережа на основі АЕ намагається під час аналізу трафіку в ІТМ визначити оптимальний підпростір, де нормальні та аномальні дані відрізняються.

Для побудови математичної моделі АЕ припустимо, що нормальний тренувальний набір є множиною $X = \{x_1, x_2, x_3, ..., x_n\}$, у якій кожен елемент є *d* розмірний вектор $(x_i \in R^d)$, а після навчання на виході АЕ отриманий результат описується множиною $\{x_1', x_2', ..., x_n'\}$. Тоді помилка реконструкції визначається як:

$$\varepsilon(x_i, x_i') = \sum_{j=1}^{d} (x_i - x_i')^2 \qquad (1)$$

Принцип виявлення вторгнень на основі АЕ полягає в тому, що звичайні дані в тестовому наборі даних відповідають нормальному профілю і відповідна помилка реконструкції є меншою, тоді як аномальні дані матимуть відносно вищу помилку реконструкції. Тому встановивши порогове значення помилки реконструкції, можна легко класифікувати аномальні дані:

$$c(x_i) = \begin{cases} normal \varepsilon_i < \theta) \\ anomalous \varepsilon_i > \theta \end{cases} \qquad (2)$$

Архітектура АЕ складається з кодера та декодера. Кодер і декодер складаються з вхідного шару нейронів, вихідного шару нейронів та одного або кількох прихованих шарів нейронів. Автоенкодлер має симетричну структуру – вихідний шар декодера дорівнює вхідному шару кодера. Математично, кодер із вхідними векторами $(x_i \in R^d)$ та вихідний шар розміру m (прихований шар) можна описати за виразом:

$$h_i = f_\theta(x_i) = s(\sum_{j=1}^{n} w_{ij}^{ex} x_j + b_i^{ex}), \qquad (3)$$

де $f_\theta(x_i)$ – функція активації вхідного шару;

x_i – вхідний вектор, $i = \overline{1,n}$;

$W^{ex} = \|w_{ij}^{ex}\|$ – матриця ваг кодера, $i, j = \overline{1,n}$;

w_{ij}^{ex} – ваги *j*-го елементу *i*-го набору даних матриці ваг кодера W^{ex}, $i, j = \overline{1,n}$;

$b^{ex} = \{b_i^{ex}\}$ – вектор зміщення, $i = \overline{1,n}$.

b_i^{ex} – зміщення *у i*-му елементі кодера, $i = \overline{1,n}$.

Відповідно до виразу (3) вхідний вектор x_i кодується у вектор меншої розмірності.

Отримане представлення h_i потім декодується назад до вихідного простору R^d за допомогою декодера, який описується наступною функцією:

$$x_i' = g_{\theta'}(h_i) = s(\sum_{j=1}^{n} w_{ij}^{вих} h_j + b_i^{вих}), \qquad (4)$$

де $g_\theta(h_i)$ – функція активації вихідного шару;

x_i' – вихідний вектор, $i = \overline{1,n}$;

θ' – множина параметрів вихідного шару $\{W^{вих}, b^{вих}\}$;

$W^{вих} = \|w_{ij}^{вих}\|$ – матриця ваг декодера;

$w_{ij}^{вих}$ – ваги *j*-го елементу *i*-го набору даних матриці ваг декодера $W^{вих}$, $i, j = \overline{1,n}$;

$b^{вих} = \{b_i^{вих}\}$ – вектор зміщення.

$b_i^{вих}$ – зміщення *у i*-му елементі декодера, $i = \overline{1,n}$.

Для мінімізації середньої помилки реконструкції будуємо наступну цільову функцію $F_{\theta',\theta'}(x_i, x_i')$ АЕ відносно параметрів θ та θ′

$$F_{\theta^*,\theta'^*}(x_i, x_i') = \arg\min_{\theta,\theta'} \frac{1}{n} \sum_{i=1}^{n} \varepsilon(x_i, x_i') = \\ = \arg\min_{\theta,\theta'} \frac{1}{n} \sum_{i=1}^{n} \varepsilon(x_i, g_{\theta'}(f_\theta(x_i))) \qquad (5)$$

де ε – функція помилка реконструкції.

Таким чином, аномальні дані можна визначити за допомогою виразу (3), але для цього потрібно визначити функції активації *f* та *g*, які мають бути нелінійними функціями, щоб виявити нелінійну кореляцію між вхідними характеристиками.

Для цього застосуємо такий метод машинного навчання без вчителя як метод ізольованого лісу [7], який може виявляти вторгнення шляхом випадкового розділення точок даних. Метод ізоляційного лісу передбачає, що дані, які не знаходяться в області їх обробки, є аномаліями. Область обробки даних формується як двійкові дерева ізоляції та ансамблі iTrees шляхом випадкової вибірки для заданого набору даних. Ключова роль дерева ізоляції полягає у виявленні аномалії для виявлення вторгнення.

Метод ізольованого лісу має декілька переваг. Спершу, для створення iTrees виконується випадковий вибір підмножини з навчального набору. По-друге, у методі iForest не використовується вимірювання відстані чи щільності для виявлення аномалії, що зменшує

витрати на обчислення порівняно з вимірюваннями відстані, задіяними в кластеризації. По-третє, метод iForest вимагає невеликої кількості пам'яті та використовує ідею ансамблю, і не залежить від того, що деякі дерева не дають ефективних результатів, оскільки алгоритми ансамблю перетворюють слабкі дерева в ефективні. Завдяки всім цим перевагам доцільно використати метод iForest для виявлення аномалій у трафіку ІТМ.

Запропонована математична модель (5) формалізує процес функціонування системи виявлення вторгнень на основі нейронної мережі, яка базується на поєднанні багатошарової згорткової нейронної мережі на основі автоенкодерів (далі – БШЗНМ МАЕ) та мережі автоенкодерів довгострокової короткочасної пам'яті (далі – МАЕ ДКП). Модель СВВ НМ обчислює показники аномалій на основі помилки реконструкції даних трафіку, що дає можливість ідентифікувати зловмисний трафік, тобто кібератаку. Ця модель виявляє вторгнення за двома послідовними діями. Тестовий набір даних надходить до БШЗНМ МАЕ, що виявляє вторгнення на основі порогового значення та розділяє вхідні дані на два набори – трафік з ознаками атаки (вторгнення) та звичайний мережевий трафік. Потім МАЕ ДКП за допомогою методу iForest визначає аномальні точки даних, тобто виявляє вторгнення.

Процес функціонування СВВ НМ доцільно розділити на такі етапи:

1. Попередня обробка даних (стандартизація та нормалізація).

2. Ідентифікація атрибутів даних мережевого трафіку на основі БШЗНМ МАЕ;

3. Розподіл мережевого трафіку на основі НМ МАЕ ДКП;

4. Виявлення вторгнення.

На першому етапі здійснюється попередня обробка даних, що можуть бути символічними та безперервними для перетворення їх в один числовий тип. Крім того, оскільки атрибути даних розподілені нерівномірно, то вони масштабуються за одним з найпоширеніших методів кодування символьних значень, що кодує числові значення на основі рівномірного розподілення в інтервалі [0–1]. Для цього використовується метод мінімально-максимальної нормалізації даних:

$$x_i' = \frac{x_i - min(x_i)}{max(x_i) - min(x_i)} \qquad (7)$$

де $\max(x_i)$ та $min(x_i)$ – максимальне і мінімальне значення вектора атрибутів x_i;

x_i' – нормалізоване значення функції між [0–1].

На другому етапі здійснюється ідентифікація атрибутів даних мережевого трафіку на основі багатошарової згорткової нейронної мережі на основі автоенкодерів. Оскільки мережевий трафік – це багатовимірний набір даних, який неможливо ідентифікувати лише за кількома окремими ознаками, то архітектуру БШЗНМ МАЕ налаштовано та трансформовано для виконання цього завдання.

Згортковий автоенкодер (далі – ЗАЕ) [10] – це особливий вид автоенкодера, який не передбачає повне підключення нейронів між собою. Модель ЗАЕ складається із конволюційних (згорткових) і деконволюційних шарів архітектури згорткової нейронної мережі (далі – ЗНМ). ЗАЕ використовує згортковий шар нейронів у частині кодера та деконволюційний шар нейронів у частині декодера. За таких умов, згортковий шар нейронів зменшує розмірність атрибутів даних, тоді як шар нейронів деконволюції збільшує розмірність цих ознак. Тобто, у ЗАЕ згортковий шар нейронів бере на себе роль кодера для виконання зменшення розмірності, тоді як шар нейронів деконволюції застосовується для реконструкції даних. Згортковий автоенкодер використовує переваги згорткового та деконволюційного шарів. Тому, порівняно зі звичайним автоенкодером, ЗАЕ має меншу кількість параметрів, тому час навчання ЗАЕ набагато менший.

Багатошарова згорткова нейронна мережа на основі автоенкодерів має кілька вузлів згортки різного розміру для отримання кількох наборів локальних функцій для досягнення точної ідентифікації. Структура БШЗНМ МАЕ базується на трьох багатошарових згортках. Багатошарові згортки обробляють набори даних за допомогою згорток розмірності 1×1, 2×2 та 3×3 для виявлення зв'язку між базовими атрибутами даних мережевого трафіку. Кодер та декодер АЕ мають вхідний шар та шар згортки, шар об'єднання, повно зв'язну нейронну мережу та вихідний шар, що має таку ж розмірність, як і вхідний шар. Для побудови математичної моделі БШЗНМ МАЕ припустимо, що $X \in \Re^{N_x \times N_y}$ і $K^f \in \Re^{a \times b}$ є вхідним вектором для ЗНМ і фільтром відповідно.

Операція згортки між вхідним вектором X і N_f фільтрами визначається таким чином:

$$Y_{i,j} = \sum_{f=1}^{N_f} \sum_{p=1}^{a} \sum_{q=1}^{b} K_{p,q}^{f} X_{i+p-1, j=q-1} \qquad (8)$$

де $Y_{i,j}$ – компоненти відфільтрованого вхідного вектору.

Розмір Y визначається його рядком Y_x і стовпцем Y_y за виразами:

$$Y_x = \frac{N_x - a + 2P}{S_x} + 1 \qquad (9)$$

$$Y_y = \frac{N_y - a + 2P}{S_y} + 1 \qquad (10)$$

де S_x і S_y – кроки в рядку та стовпці відповідно, які керують зміщенням фільтра на вхідних даних;

P – відступ, який контролює кількість нулів навколо X. Відступ використовується для зміни розміру виходу ЗНМ без шкоди для результату згортки.

Припускаючи $d = N_x \times N_y$, можна відобразити

будь-яку точку даних x_k до точки $x_{i,j}$ у двовимірному масиві (який виглядає як $N_x \times N_y$ матриця). Тобто

$$x_k \equiv X_{i,j},$$

$$k = \overline{1,d},\ i = \overline{1,N_x} \text{ та } j = \overline{1,N_y};$$

де $$h_l = \sigma\left(\sum_{k=1}^{d'} w_{lk}^{'} x_{\phi(k)} + b_l\right) \quad (11)$$

$$d' = Y_x \times Y_y;$$

$$w_{lk}^{'} = \sum_{k=1}^{N_f}\sum_{p=1}^{a}\sum_{q=1}^{b} K_{p,q}^{f} w_{lk} \quad (12)$$

та

$$x_{\phi k} \equiv X_{i+p-1,\,j+q-1}$$

де $\phi(k) = (i+p-1),\ \ j+q-1$;

$w_{ik}^{'}(\forall l,k)$ – нові латентні просторові ознаки, які є вхідними даними до МАЕ ДКП.

На третьому етапі здійснюється розподіл мережевого трафіку з використанням НМ на основіф автоенкодерів довгострокової короткочасної пам'яті. Автоенкодер в МАЕ ДКП складається з кодера та декодера. Завданням кодера є вивчення основних характеристик і створення закодованої версії вхідного зразка, а декодер – реконструкція вхідних даних.

Функцією кодера архітектури МАЕ ДКП є перетворення послідовності латентних просторових ознак, що були витягнуті з мережевого трафіку за допомогою БШЗНМ МАЕ в фіксований вектор нових ознак (латентний простір), який, в свою чергу, декодером перетворюється на вихідну послідовність. Така конфігурація АЕ здатна виявляти короткі та довгі залежності в послідовності базових ознак.

МАЕ ДКП може запам'ятовувати довготривалі залежності у ДКП-комірках – c_i. Комірки c_i оновлюються за допомогою чотирьох внутрішніх активаційних шарів (гейтів), які є нейронними шарами сигмоїдної нейронної мережі, із виконанням покомпонентної операції над ними.

Кожен гейт призначений для виконання окремої функції:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (13)$$

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (14)$$

$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C) \quad (15)$$

$$O_t = \sigma(W_O[h_{t-1}, x_t] + b_O) \quad (16)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (17)$$

$$h_t = O_t * \tanh(C_t) \quad (18)$$

де W_f, W_i, W_C, W_O – лінійні перетворення;

C_t та h_t – пам'ять комірки та вихідне значення відповідно в момент часу t.

На виході МАЕ ДКП маємо послідовність трафіку $X(n) = [x^{(1)}, x^{(2)}, \ldots, x^{(W)}]$ довжиною W, де n - індекс звичайного тренувального прикладу.

Кодер АЕ ДКП генерує синтезований вихідний вектор (y^W) з попередньо визначеною розмірністю $r \times 1$ на основі рівнянь (13)-(18):

$$y^W = \psi[x^{(1)}, x^{(2)}, \ldots, x^{(W)}] \quad (19)$$

де ψ – нелінійна функціюя кодера архітектури ДКП.

Вектор (y^W) є новими латентними просторовими ознаками, які виражають компактне представлення поведінки у часі базових ознак. Вектор (y^W) використовується декодером для відновлення вхідного зразка відповідно до виразу (19):

$$\hat{X}(n) = \Phi[y^{(1)}, y^{(2)}, \ldots, y^{(r)}] \quad (20)$$

де Ф – функція декодера МАЕ ДКП.

Метою декодера є відновлення вхідної послідовності з мінімальною втратою, що може бути обчислена з урахуванням середньоквадратичної помилки відповідно до (1).

Четвертий етап передбачає виявлення вторгнення. Оскільки АЕ навчається тільки на «нормальних» даних, тому втрати реконструкції для даних атаки є набагато вищими, ніж для «нормальних» даних, тобто:

$$Y_i = ((Y_P^{/}), (Y_q^{/})) \quad (21)$$

де $(Y_P^{/})$ – вектор «нормального» пакету даних, у яких помилка реконструкції менша, ніж граничне значення;

$(Y_q^{/})$ – вектор даних з вищою помилкою реконструкції, які вважаються «атаками».

Оскільки результат АЕ не є стовідсотково точним, як $(Y_P^{/})$, так і $(Y_q^{/})$ містять як дані про атаку, так і нормальні дані відповідно.

Для досягнення більшої точності, тобто виявлення більшої кількості вторгнень, ці два набори подаються на вхід двох модулів *iForest*. Перший модуль *iForest-1* отримує результати «атак» від АЕ і шукає нормальні точки даних. Другий модуль *iForest-2* бере вихід «нормальних» даних від АЕ і шукає атаковані точки даних. Дані атаки у наборі «нормальних» і нормальні дані у наборі «атак» є ніщо інше, як викиди або аномалії.

Модуль *iForest-2* бере «нормальний» набір $(Y_P^{/})$ і шукає дані про атаку. Оскільки АЕ вже ідентифікував більшість нормальних і атакованих пакетів на першому етапі, набір $(Y_P^{/})$ містить меншу кількість атакованих пакетів.

Набір $(Y_q^{/})$, що містить «атаковані» дані, подається на вхід *iForest-1*. $(Y_q^{/})$ також містить деякі фактичні нормальні дані. *iForest-1* шукає ці «викиди» у $(Y_q^{/})$, тобто

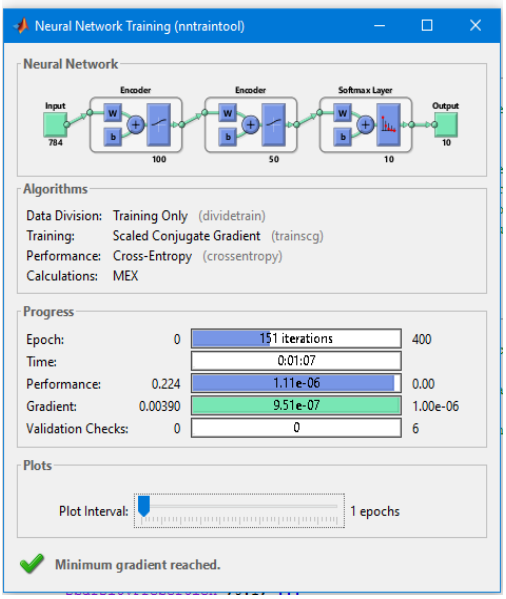
$$Y_P, O_q \leftarrow iForest - 1((Y_P^{/})) \quad (22)$$

$$Y_q, O_p \leftarrow iForest - 2((Y_q')) \quad (23)$$

У кінцевому підсумку, (Y_p, O_q) і (Y_q, O_p) є остаточними наборами нормальних і шкідливих пакетів мережевого трафіку.

Toolbox програмного середовища MATLAB та натреновано з використанням набору даних UNSW-NB15. Результати тренування наведено на рисунку 1.

Математичну модель СВВ НМ було побудовано за допомогою пакету програм Neural Network.



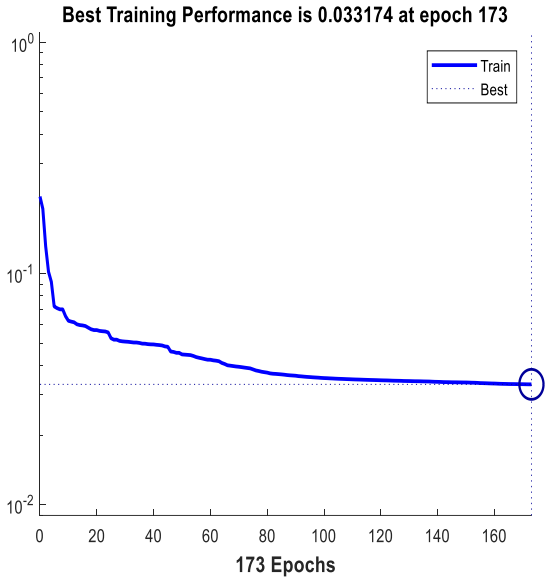


Рисунок 1 – Модель СВВ НМ та результати її тренування

Висновки й перспективи подальших досліджень

У цій статті представлено розроблену математичну модель системи виявлення вторгнення з використанням нейронної мережі на основі автоенкодерів, яка формалізує процес виявлення вторгнень у інформаційно-телекомунікаційних мережах військового призначення шляхом виявлення взаємозалежності між базовими атрибутами мережевого трафіку.

Системи виявлення вторгнень нейронних мереж поєднує в собі багатошарову згорткову нейронну мережу на основі автоенкодерів та мережу автоенкодерів з використанням довгострокової короткочасної пам'яті для виявлення просторово-часової залежності в даних мережевого трафіку. Продуктивність запропонованого підходу було оцінено з використанням набору даних UNSW-NB15.

Напрямом подальших досліджень є розроблення математичної моделі виявлення вторгнень на основі автокомпенсаційного принципу з використанням запропонованої в цій статті математичної моделі системи виявлення вторгнень на основі автоенкодерів.

Список бібліографічних посилань

1. Ahmad M., Basheri M. J. Iqbal, Rahim A. Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection, *IEEE Access*. 2018. № 6. 33789–33795. doi:10.1109/ACCESS.2018.2841987. **2. Auskalnis J., Paulauskas N., Baskys A.**, Application of local outlier factor algorithm to detect anomalies in computer network. *Elektronika ir Elektrotechnika*. 2018. №24(3). С. 96–99, cited By :2. **3. Rathore S., Park J. H.** Semi-supervised learning based distributed attack detection framework for iot, *Applied Soft Computing Journal*. 2018. № 72. С. 79–89, cited By :80. **4. Aliakbarisani R., Ghasemi A., Felix Wu S.** A data-driven metric learningbased scheme for unsupervised network anomaly detection. *Computers and Electrical Engineering*. 2019. №73. С. 71–83, cited By :7. **5. Karami A.,** An anomaly-based intrusion detection system in presence of benign outliers with visualization capabilities, *Expert Systems with Applications*. 2018. № 108. С. 36–60, cited By :28. **6. Alom M. Z., Bontupalli V., Taha T. M.** Intrusion detection using deep belief networks, in: 2015 *National Aerospace and Electronics Conference* (NAECON), 2015, pp. 339–344. doi:10.1109/NAECON.2015. 7443094. **7. Kang M.-J., Kang J.-W.** Intrusion detection system using deep neural network for in-vehicle network security. *PloS one*. 2016. № 11 (6). e0155781. **8. Gao N., Gao L., Gao Q., Wang H.** An Intrusion detection model based on deep belief networks, in: 2014 *Second International Conference on Advanced Cloud and Big Data, IEEE*. 2014. P. 247–252. **9. Zhang X., Chen J.** Deep learning based intelligent intrusion detection, in: 2017 IEEE 9th International *Conference on Communication Software and Networks (ICCSN), IEEE*. 2017. P. 1133–1137. **10. Yu Y., Long J., Cai Z.** Network intrusion detection through stacking dilated convolutional autoencoders, Security and Communication Networks 2017.

MATHEMATICAL MODEL OF AN AUTOENCODER FOR ENSURING CYBERSECURITY OF MILITARY INFORMATION AND TELECOMMUNICATIONS NETWORK

Ilyin Dmytro
Starinskyi Ivan ***(Candidate of technical sciences, senior researcher)***

National Defence University of Ukraine, Kyiv, Ukraine

The article demonstrates that the traffic of military-purpose Information and Telecommunication Networks (ITN) encompasses a substantial volume of data sets. Ensuring the security of military ITN against cyberattacks is a highly labor-intensive and error-prone process. In this context, network traffic data exhibit intricate nonlinear relationships that evolve over time. Existing cybersecurity models are based on data correlation models for traffic. These models often demand significant computational resources and do not permit real-time network traffic processing, neglecting spatiotemporal correlations within the data. To address this issue, a unified autoencoder, named Multi-Scale Convolutional Neural Network-Long Short-Term Memory Autoencoder (MSCNN-LSTM-AE), is proposed for anomaly detection in network traffic. The model initially employs a multi-scale convolutional neural network autoencoder (MSCNN-AE) to analyze the spatial features of the data set. Subsequently, the latent space features obtained from the MSCNN-AE are utilized in an autoencoder network based on Long Short-Term Memory (LSTM) for anomaly detection in network traffic. The model also employs two Isolation Forest algorithms as error correction mechanisms to address false positives and false negatives, thus enhancing detection accuracy. The evaluation of the NSL-KDD and UNSW-NB15 models on the CICDDoS2019 dataset indicates that the proposed mathematical model significantly outperforms existing mathematical models.

Keywords: *cybersecurity, cyberattack, cyber incidents, cyber threats, information security, cybersecurity strategy.*

References

1. Ahmad, M., Basheri, M. J. Iqbal, Rahim, A., (2018). Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection, IEEE Access 6 33789–33795. doi:10.1109/ACCESS.2018.2841987. **2. Auskalnis, J., Paulauskas, N., Baskys, A.,** (2018). Application of local outlier factor algorithm to detect anomalies in computer network. *Elektronika ir Elektrotechnika*, 24 (3), 96–99, cited By: 2. **3. Rathore S., Park J. H.,** (2018). Semi-supervised learning based distributed attack detection framework for IoT, *Applied Soft Computing Journal*, 72, 79–89, cited By: 80. **4. Aliakbarisani, R., Ghasemi, A., Felix Wu, S.,** (2019). A data-driven metric learning-based scheme for unsupervised network anomaly detection. *Computers and Electrical Engineering*, 73, 71–83, cited By: 7. **5. Karami, A.,** (2018). An anomaly-based intrusion detection system in presence of benign outliers with visualization capabilities. *Expert Systems with Applications* 108 36–60, cited By: 28. **6. Alom, M. Z., Bontupalli, V., Taha, T. M.,** (2015). Intrusion detection using deep belief networks. In: 2015 *National Aerospace and Electronics Conference (NAECON)*, 339–344. doi:10.1109/NAECON.2015.7443094. **7. Kang M.-J., Kang J.-W.,** (2016). Intrusion detection system using deep neural network for in-vehicle network security. *PloS one*, 11(6), e0155781. **8. Gao, N., Gao, L., Gao, Q., Wang, H.,** (2014). An intrusion detection model based on deep belief networks. In: 2014 *Second International Conference on Advanced Cloud and Big Data, IEEE*, 247–252. **9. Zhang, X., Chen, J.,** (2017). Deep learning based intelligent intrusion detection. In: 2017 IEEE *9th International Conference on Communication Software and Networks (ICCSN), IEEE*, 1133–1137. **10. Yu, Y., Long, J., Cai, Z.** Network intrusion detection through stacking dilated convolutional autoencoders. *Security and Communication Networks*, 2017.