УДК 004.492, 004.89

Б.О. Панчук

ГЕНЕРАЦІЯ ТА ВИКОРИСТАННЯ ЗМАГАЛЬНОЇ ВИБІРКИ ДЛЯ ПРОТИДІЇ УХИЛЕННЮ БОТНЕТІВ ВІД ВИЯВЛЕННЯ НЕЙРОННИМИ МЕРЕЖАМИ

(до 100-річчя з дня народження академіка В.М. Глушкова)

Панчук Богдан Олександрович

Інститут кібернетики імені В.М. Глушкова НАН України, м. Київ,
orcid: 0000-0002-5389-359X

bogdanscloud@gmail.com

Описано спосіб проведення оцінки надійності систем виявлення ботнетів на основі нейронних мереж, з точки зору їх вразливості до змагальних атак, і підвищення стійкості таких систем через розширення навчального набору штучними даними. Під «змагальною атакою» мається на увазі цілеспрямована спроба зловмисника штучно модифікувати дані (у даному разі — потоки мережевих пакетів) для спроби ухилення від виявлення класифікатором. Запропонований та реалізований метод генерації змагальних прикладів для системи класифікації трафіку на основі нейронної мережі шляхом адаптації «методу швидкого знаку градієнта», відомого з області обробки зображень, для роботи з мережевими даними, представленими у формі мережевих потоків. Характерними рисами описаного підходу є обчислювальна простота, а також правдоподібність отриманих прикладів трафіку. Правдоподібність штучно створених екземплярів потоків забезпечувалась накладанням глобальних споріднених груп ознак, над якими проводились модифікації. Окрім застосування описаного підходу, для оцінки вразливості моделей класифікації також показана можливість його застосування для доповнення початкової навчальної вибірки штучними даними. Спершу базова модель класифікації була навчена на відкритому наборі даних трафіку ботнетів. Далі початковий набір був розширений змагальними прикладами, згенерованими описаним методом, після цього було експериментально показано, що модель, навчена на розширених даних, більш стійка до змагальних атак порівняно з базовою. При цьому метод не є специфічним лише для області виявлення ботнетів, а може застосовуватися для мережевих атак іншого роду за умови наявності відповідного навчального набору. Запропоновано напрямки для подальшого розвитку цього підходу.

Ключові слова: мережеві потоки, нейронні мережі, системи виявлення вторгнень, змагальні атаки, доповнення даних, ботнети.

Вступ

Засновник сучасної алгебраїчної школи та ініціатор перших досліджень зі штучного інтелекту (ШІ) в Україні, академік Віктор Глушков, передбачив широке використання методів ШІ, наприклад область кібербезпеки, яка є актуальним на-

прямком, особливо під час військових дій. В останні роки дослідженням кібербезпеки із застосуванням ШІ приділено значну увагу [1]. Інтерес обумовлений помітною слабкістю більш конвенційних методів, таких як співставлення сигнатур поведінки, для виявлення загроз, де поведінка є складною та непередбачуваною. Одним із найбільш яскравих представників таких загроз є ботнети. Ботнет — це мережа вузлів, заражених злоякісним програмним забезпеченням, яка контролюється зловмисником для проведення різного роду координованих атак, наприклад DDoS (distributed denial-of-service). Перспективним напрямком протидії такій загрозі є аналіз та класифікація мережевих даних моделями ШІ для виявлення діяльності ботнетів, на цю тему написано багато наукових праць [2]. Автори наводять високі класичні показники ефективності моделей класифікації, отримані на використаних ними наборах мережевих даних. Однак при більш глибокому дослідженні помітно, що в багатьох таких роботах є спільні й часто розповсюджені недоліки, які здебільшого спричинені нестачею навчальних даних. Саме тому підходи до виявлення мережевих вторгнень методами ШІ здебільшого так і залишаються експериментальним і не набувають широкого використання.

У відкритому доступі існує невелика кількість розмічених наборів мережевих даних діяльності ботнетів, які можна було б використовувати для навчання та оцінки якості моделей класифікації, а ті, що є у наявності, часто застарілі та недостатньо репрезентативні. Це породжує щонайменше дві проблеми:

1) висока повторюваність поведінок ботнетів у навчальному та тестувальному наборах даних ставить під сумніви якість генералізуючих властивостей моделі класифікації; немає гарантій, що при внесенні незначних відмінностей у вигляд трафіку модель зможе розпізнати атаку (навмисні маніпуляції зі сторони хакера, зміни мережевої топології чи просто випадкові збурення);

2) асиметрія навчальних наборів даних через відмінність у відношенні об’єму злоякісного трафіку до доброякісного призводить до високої частки хибнопозитивних результатів при класифікаціх, що є критичним показником при застосуванні в інформаційній безпеці.

Окремою, ще більш загальною проблемою є вразливість класифікаторів до «змагальних атак» [3]. Змагальна атака — це специфічний феномен можливості штучного створення екземплярів вхідних даних, які модель систематично відноситиме до конкретного класу за бажанням зловмисника, незалежно від того, якому класу вони належать в дійсності. Хоча ця риса і притаманна моделям ШІ у будь-яких областях використання, та особливо гостро вона проявляється в області виявлення мережевих атак. При проєктуванні будь-якої системи виявлення вторгнень апріорі очікується, що і вона сама може стати ціллю атаки. Відповідно від неї вимагається значно вища стійкість до різних методів ухиляння від виявлення. Проблема нестачі навчальних даних, згадана вище, може призвести до перенавчання («overfitting») класифікатора [4], а також погіршити його екстраполюючу здатність із-за властивості кускової лінійності нейронних мереж відносно вхідних даних [5]. Обидва фактори роблять моделі ШІ ще вразливішими до змагальних атак.

Зауважимо, що найбільш розповсюджені оцінки якості класифікації, такі як «точність» та «площа під ROC-кривою», які зазвичай наводяться в роботах на дану тему [6–8], самі по собі не відображають стійкість класифікаторів до змагальних атак, і отримані моделі потребують додаткових гарантій якості. Саме тому необхідні додаткові кроки для покращення генералізуючих властивостей моделей та підкріплення достовірності отриманих оцінок. Одним з напрямків можливого покращення є розширення оригінальної вибірки штучно згенерованими, зокрема

змагальними, екземплярами і подальше тестування на стійкість до змагальних атак. Очікується, що при отриманні достатньо якісного розширеного набору даних оцінки ефективності моделі на ньому будуть більш надійними.

Мета даної публікації — розробка методу розширення початкової навчальної вибірки даних для моделей виявлення ботнет-трафіку штучними, але правдоподібними даними, з метою підвищення стійкості моделей до змагальних атак, які навчатимуться на даному наборі. Крім того, ставилось кілька критеріїв до розробленого методу. По-перше, необхідно врахувати семантику мережевих потоків та забезпечити її збереження при генерації нових даних. По-друге, трансформації проведені над початковими даними для створення нових, повинні інтерпретуватися і бути реалістичними з точки зору зловмисника. По-третє, необхідно переконатися, що отримане розширення даних дійсно підвищує стійкість навчених на ньому моделей виявлення до змагальних атак, і при цьому не спричиняє регресію їх базових показників якості.

У даній роботі описується метод генерації змагальних екземплярів мережевих потоків, які можуть відображати потенційні маніпуляції зловмисника над трафіком з метою введення в оману нейронну мережу, що проводить класифікацію. Реалізовано адаптацію методу FGSM (вперше застосовано в області класифікації зображень [5]) для роботи з трафіком представленим мережевими потоками. Змагальна вибірка створювалась супроти нейронної мережі-класифікатора, навченої на відкритому наборі даних CIC Botnet 2014 [9], який неодноразово використовувався в різних працях для виявлення ботнетів методами ШІ [7–9]. Наведено статистичні дані з якості отриманих екземплярів і оцінено вразливість моделі до змагальних атак такого роду. Далі нову модель класифікації було навчено на доповненому змагальною вибіркою наборі даних і підраховано нові показники точності та вразливості. Після цього наведено порівняння отриманих показників якості моделі до й після, і в кінці зроблено висновки про ефективність застосування даного методу та подальші напрямки розвитку.

Зауважимо, що хоча в рамках цієї роботі у ролі об’єкта дослідження взято проблему виявлення нейронними мережами даних ботнетів, однак описану методику також можна узагальнити для виявлення мережевих атак інших типів.

Виділення мережевих потоків та їх класифікація

У будь-якій системі виявлення вторгнень (Intrusion Detection System — IDS), задача виявлення зводиться до спостереження за певними подіями і класфікації цих подій чи їх послідовностей на доброякісні чи злоякісні. У сфері протидії мережевим атака об’єктом класифікації є мережевий трафік, представлений у вигляді перехоплених пакетів.

У «докладному аналізі» окремих пакетів (Deep Packet Inspection — DPI, перевірка та фільтрація пакетів за змістом) є переваги і недоліки, а саме, складність відновлення семантичного контексту, необхідність аналізу часових рядів, та неможливість аналізу зашифрованих даних.

Щоб уникнути згаданих недоліків, у даній роботі використано альтернативний підхід: агрегація пакетів у мережеві потоки — логічні групи індивідуальних пакетів, які представляють завершений «діалог» між двома віддаленими процесами. Наприклад, потік може складатися з усіх пакетів в рамках одного TCP-з’єднання. Групування проводиться за кортежем *<IP-адреса джерела, IP-адреса призначення, Порт джерела, Порт призначення, Протокол, Часова мітка початку>*.

Далі кожен потік представляється вектором ознак, які його характеризують. Ознаки є статистиками, підрахованими з пакетів, які належать даному потоку. Після

цього проводиться нормалізація значень і вектори ознак передаються нейронній мережі для її навчання чи класифікації мережевого потоку на доброякісний чи злоякісний.

Повна схема класифікації мережевого трафіку на основі потоків показана на рис. 1. Детальний опис алгоритму виділення потоків та порівняльна характеристика ефективності різних класифікаторів при використанні такого методу наведено в [10].

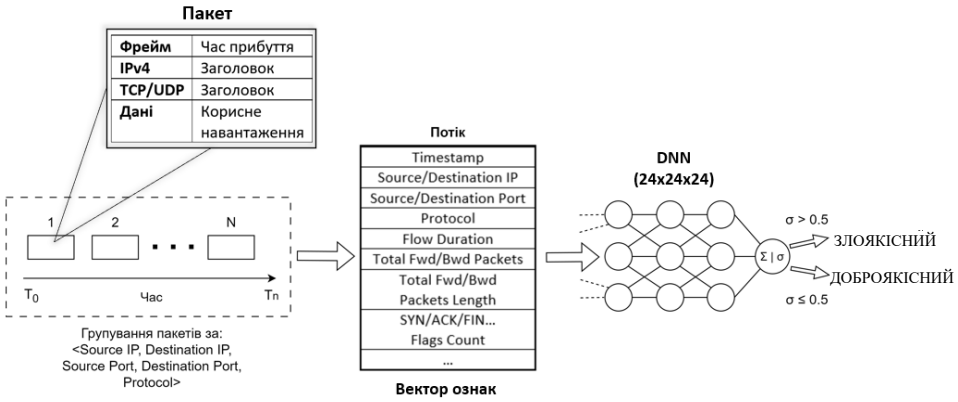

Рис. 1

Доповнення даних

При доповненні даних початкова вибірка розширюється новими штучно створеними елементами, які в подальшому можна застосовувати для навчання чи тестування моделі. Для цього до початкових екземплярів реальних даних приміряється ряд модифікацій, таким чином з одної точки в початковому просторі даних отримується кілька нових. При цьому нові екземпляри мають бути правдоподібними, тобто достатньо схожими на оригінали, зберігаючи семантику даних.

Добре відомі методи доповнення даних в більш досліджених областях, таких як обробка зображень, аудіо та тексту. Однак для обробки мережевих потоків застосування класичних методів обмежене через особливості представлення та семантику даних такого роду. На відміну від області обробки зображень, генерація нових мережевих даних через накладання шумів обмеженої інтенсивності виявилось малоефективною. При класифікації потоків з накладеним шумом, амплітуда якого не перевищує 10 % від оригінального значення, погіршення точності класифікації склало менше 0,01 %, а значить, модель досить стійка до випадкових збурень.

Значно вищого відсотку міскласифікацій можна досягти, використовуючи не випадкові шуми, а спеціальні модифікації, отримані методами з вирахування «змагальних прикладів» — термін вперше запропонований в [3].

Генерація змагальних прикладів

У цьому розділі описана адаптація «швидкого методу знаку градієнта» (FGSM) [5] для перетворення мережевих потоків, представлених векторами ознак. Ціль перетворення: із початково правильно класифікованих нейронних мереж зробити семантично схожі екземпляри, але такі, що модель класифікуватиме їх невірно. У подальшому, розширивши навчальну множину екземплярами такої природи, можна отримати більш репрезентативну вибірку. Очікується, що навчена на ній модель буде більш стійкою до схожих збурень у вхідних даних.

Для мережевих потоків алгоритм генерації буде таким: обираються потоки, які належать заздалегідь відомому класу (наприклад, «злоякісні»). Нехай потік

має N ознак та позначається $x = (x_1, \ldots, x_N)$. Відносно кожного x знаходиться градієнт ∇_x функції втрат $J(\theta, x, y)$, яка використовувалась при навчанні моделі. Функція втрат J — це бінарна перехресна ентропія, обчислена для даного x, та мітки його фундаментальної істини y (мітка 1 позначає потік як злоякісний, 0 — як доброякісний), θ — це ваги та зсуви моделі, вивчені нею під час навчання.

Знаходження градієнта $\nabla_x J(\theta, x, y)$ відбувається через обчислення частинних похідних J по кожній з компонент (ознак) x:

$$\nabla_x J(\theta, x, y) = \left(\frac{\partial J(\theta, x, y)}{\partial x_1}, \ldots, \frac{\partial J(\theta, x, y)}{\partial x_N} \right).$$

Частинні похідні нелінійної функції, якою є J, обчислюються стандартним алгоритмом зворотного поширення [11] (окремий випадок автоматичної диференціації, який реалізується шляхом зворотної акумуляції в обчислювальному графі [12]).

1. Дано елемент вибірки x. Спершу проводиться прямий прохід графом з обчисленням результату кожної операції для даного x.

2. Далі, починаючи з вихідного шару і рухаючись у зворотному напрямку, шар за шаром, обчислюються частинні похідні функції втрат за параметрами та операціями, представленими даним шаром. Обчислення відбувається за класичним ланцюговим правилом повної диференціації функції.

3. Нарешті, коли процедура зворотного розповсюдження доходить до першого «вхідного» шару, частинні похідні $\frac{\partial J(\theta, x, y)}{\partial x_i}$ обчислюються аналогічно похідним за всіма іншими параметрами даного обчислювального графу.

Отриманий градієнт вказує напрямок найбільш стрімкого зростання значення функції втрат J у конкретній точці простору ознак. Чим більша величина i-ї компоненти градієнта, тим чутливіша модель класифікації до локальної зміни i-ї ознаки (x_i).

Оскільки трансформації у просторі ознак не завжди мають зворотне відображення у простір подій оригінальної задачі, може виникати необхідність накласти певні обмеження на припустимі зміни в ознаках екземпляра. Позначимо $\overline{\nabla_x}$ вектор попередньо обчислених компонентів градієнта, де певні обрані ознаки залишаються незмінними, а всі інші виставляються в 0 (іншими словами, ігноруються). Тоді для отримання потенційного змагального прикладу ознаки оригінального екземпляру необхідно змасштабувати в напрямку підрахованого градієнту, таким чином збільшуючи значення функції втрат. Отримання нового екземпляра x' з оригінального x показано нижче:

$$x' = x * (1 + \varepsilon * sign(\overline{\nabla_x} J(\theta, x, y))), \varepsilon \in (0, \ldots, 1). \quad (1)$$

Зауважим, що в класичному FGSM [5] використовується формула, яка передбачає, що ознаки x_i попередньо змасштабовано (як правило, відносно мінімального та максимального значень, що зустрічались у навчальному наборі), і її значення тепер лежить в межах від [0, ..., 1]. При цьому для кожного оброблюваного прикладу значення відповідної ознаки збурюється на певну абсолютну величину. Якщо нове значення ознаки x_i' виходить за зазначені межі, то воно «обрізається», тобто встановлюється в 0 чи 1:

$$x' = x + \varepsilon * sign(\overline{\nabla_x} J(\theta, x, y)), \varepsilon \in (0, ..., 1), x_i \in [0, ..., 1], x'_i \in [0, ..., 1], \quad (2)$$

де x_i та x'_i — i-та ознака початкового й згенерованого прикладу відповідно (2).

При виконанні цієї роботи помічено, що спроба генерації нових мережевих потоків формулою (2) призводить до двох проблем.

1. Якщо діапазон масштабування ознаки, отриманий з навчального набору, досить великий, то навіть при відносно невеликих ε згенерований потік легко виходить за рамки семантичної подібності. Припустимо, що значення «Розмір корисного навантаження» для конкретного потоку x_i = 1010 байт, але діапазон значень для цієї ознаки, в межах якого проводилось масштабування, скажімо, [0, ..., 10000], тоді при ε = 0,1 і знаку граідєнта «–» значення ознаки, отримане з формули (2), матиме вигляд $x'_i = \frac{1010}{10000} + 0{,}1 * (-1) = 0{,}101 - 0{,}1 = 0{,}001$, що відповідає лише 10 байтам. Таке різке зменшення розміру корисного навантаження (більше, ніж в 100 разів) порушує вимогу подібності згенерованого потоку до оригінального.

2. Крім того, значно зростає кількість випадків, коли обрізається значення ознаки. Фактично при додаванні абсолютної величини «ε» будь-яке значення ознаки, яке лежить в межах [0, ..., ε] і [ε, ..., 1], буде обрізане при знаках градієнта «–» та «+» відповідно. При обробці мережевих потоків, представлених лише кількома десятками ознак, втрата інформації по одній з них досить критична.

Заміна додавання однакового абсолютного значення до ознаки для всіх прикладів (2) на відносне масштабування її величини для кожного окремого прикладу (1) повністю вирішує проблему 1 і також запобігає обрізкам значень по нижній межі, при ε < 1, і зменшує частоту обрізок по верхній межі, що частково вирішує проблему 2.

Особливості простору ознак мережевих потоків

Обробка мережевих даних у формі статистик мережевого потоку має певну специфіку, без врахування якої спроби генерації нових правдоподібних потоків будуть невдалими.

По-перше, простір ознак мережевих потоків низьковимірний, на що зверталась увага в попередньому розділі. Якщо зображення складається з мільйонів пікселів, потік, як правило, представляється лише кількома десятками значень. В експериментальній частині даної роботи потоки представлялися 50 ознаками (Додаток).

По-друге, більшість ознак не мають чітко встановленої верхньої межі. На відміну від зображень, де кожен піксель закодований фіксованою кількістю бітів і кольори пікселів достатньо рівномірно розподілені по всьому спектру, значення характеристик потоку можуть необмежено зростати і значно варіюватися, адже їх кодування ніяк не регламентує їх верхню межу. Прикладом є ознака «кількість пакетів у потоці» та інші, споріднені їй. Навіть для тих характеристик, для яких існують певні технічні обмеження, верхня межа варіюється від мережі до мережі, і її можна оцінити хіба що статистично. Наприклад, максимальне значення «тривалість потоку» може по-різному обмежуватися залежно від конфігурації мережевого шлюзу чи веб-сервера.

По-третє, ознаки потоків зазвичай дуже гетерогенні. Вони значно відрізняються одна від одної верхніми та нижніми межами, а також дисперсією. Наприклад, значення «кількість TCP ACK прапорців», як правило, суттєво менше за «максимальний розмір корисного навантаження» в одному й тому ж потоці. Це також є значною відмінністю від області обробки зображень, де ознаки фактично гомогенні.

Крім того, не всі ознаки доступні для довільної модифікації без спричинення порушення семантики потоку. Такі поля, як «кількість прапорців SYN/ACK/ FIN…», випливають з правил, встановлених TCP-протоколом.

Також через специфіку підрахунку та представлення ознак потоку (більшість з них це статистики) деякі ознаки корелюють та динамічно обмежують одна одну. Наприклад, коли величини описують мінімальне, максимальне та сумарне значення певного параметра, то, принаймі, має виконуватись нерівність: «мін.» ≤ ≤ «макс.» ≤ «сумарна».

Звідси зрозуміло, що генерація довільних варіацій мережевих потоків (не лише змагальних), як і взагалі будь-яка контрольована генерація мережевого трафіку, є нетривіальною задачею і потребує окремого дослідження.

При цьому для генерації саме змагальних прикладів варто враховувати, що зловмисник має можливість змінювати значення ознак потоку лише опосередковано — величини ознак потоків обчислюються системою аналізу мережевого трафіку, до якої зовні немає прямого доступу, що ускладнює завдання створення успішних атак проти класифікатора. Водночас підвіщення стійкості до таких атак може зводитися до створення надмножини, яка включає й певний відсоток реально недосяжних екземплярів доти, доки ця множина максимально широко покриває всі реально можливі атаки. Саме цей метод затосовано в практичній частині, описаній нижче. Для мінімізації ризику порушення семантики потоку при модифікації та виходу за межі досяжності збурення були здійснені лише над обмеженими групами ознак, в яких вони мали б конкретну семантичну інтерпретацію. При цьому було накладено низку базових обмежень, при порушенні яких згенерований екземпляр відкидався.

Набори даних та ознаки

Для отримання практичних результатів використовувався набір даних UNB CIC Botnet 2014 [9]. Самі потоки та їх ознаки із набору пакетів виділені модифікованою версією інструменту CICFlowMeter [13]. Окрім стандартного ідентифікуючого кортежу, який складається з IP-адрес, портів, протоколу та часової мітки, для кожного потоку виділено 50 ознак (див. Додаток).

Зауважимо, що ботнети поділяються на три основні типи за способом комунікації між різними вузлами їх робочої ієрархії [14]: IRC (Internet Relay Chat), HTTP (HyperText Transfer Protocol) та P2P (peer-to-peer). Оскільки у навчальному наборі даних повністю відсутні приклади P2P-трафіку, то при оцінюванні якості моделі на тестувальному наборі для отримання більш чітких результатів потоки P2P-ботнетів також виключались.

Кількість потоків за класами в навчальному та тестувальному наборах наведено у табл. 1. Для навчальних даних приклади, отримані при різних значеннях ε, об’єднувалися в спільну множину, яка доповнювала початковий навчальний набір. Для тестувальних даних показники класифікації підраховувались для кожного значення ε індивідуально, тому в табл. 1 їх кількість не наводиться. Під тестувальними даними мається на увазі вибірка, з якою модель не зустрічалась при навчанні (ні самі елементи вибірки, ні похідні від них).

Таблиця 1

Дані	**Навчальні**		**Тестувальні (нові)**	
Доповнення	**До**	**Після**	**До**	**Після**
Доброякісні	306056	1191153	126232	–
Злоякісні (ботнети)	109305	396711	87960	–
% злоякісних	35,7	33,3	69,7	–

Для отримання більш правдоподібних прикладів застоування збурень та подальша оцінка стійкості моделі проводилась лише за двома семантично виокремленими групами ознак.

• Група ознак 1 (9 ознак): тривалість потоку, мінімальний, максимальний міжпакетний інтервал в прямому та зворотному напрямках, середньоквадратичне відхилення міжпакетних інтервалів у прямому та зворотному напрямках, сума міжпакетних інтервалів в обох напрямках.

• Група ознак 2 (6 ознак): мінімальне, максимальне та середньоквадратичне відхилення довжини пакетів у прямому та зворотному напрямках.

Збурення в ознаках «групи 1» можна інтерпретувати як маніпуляцію з часовими параметрами потоку: віддалення чи зближення першого та остатнього пакетів визначає значення «тривалості потоку», а додаткова затримка (чи, навпаки, її скорочення) між відправкою двох послідовних пакетів впливає на статистику міжпакетних інтервалів.

Зміни в ознаках «групи 2» відображають варіювання розмірів пакетів. Наприклад, додаючи надлишкові байти в кінець корисного навантаження, можна збільшити статистику «максимальної довжини пакета», а організувавши дані більш компактно (чи взагалі передавши пустий пакет), зменшити значення «мінімальної довжини».

При цьому, хоча й при внесенні довільних змін в ознаки кожної з груп не можна гарантувати повну семантичну тотожність нового отриманого потоку, однак, як буде показано експериментально, генерація розширення даних з додаванням лише базових логічних обмежень достатня для якісного покращення стійкості моделі класифікації.

Для кожної з груп ознак введені обмеження (повний список назв ознак та їх опис наведено у Додатку).

Група 1:

«Fwd IAT Min» ≤ «Fwd IAT Max»
«Bwd IAT Min» ≤ «Bwd IAT Max»

Група 2:

«Fwd Packet Length Min» ≤ «Fwd Packet Length Max»
«Bwd Packet Length Min» ≤ «Bwd Packet Length Max»
«Fwd Packet Length Max» ≤ «Total Length of Fwd Packet»
«Bwd Packet Length Max» ≤ «Total Length of Bwd Packet»

Базова модель та оцінка показників ефективності та стійкості

Оскільки навчання моделі і пошук змагальних прикладів є ресурсоємними задачами, а проведені експерименти з доповнення даних потребували багаторазових повторень, то одна з основних вимог до моделі — обчислювальна простота. Крім того, використання відносно простої архітектури нейронної мережі дозволяє зменшити ризик отримання вдалих, але невідтворюваних результатів через випадкову успішну ініціалізацію початкових вагових коефіцієнтів класифікатора при навчанні.

Як модель класифікації було застосовано глибоку нейронну мережу з трьома прихованими повнозв'язними шарами по 24 нейрони та одним вихідним. У прихованих шарах у ролі функції активації використовується ReLU (зрізаний лінійний вузол), адже знаходження градієнта для неї тривіальне (частинна похідна завжди 0 або 1). Як функція активації вихідного нейрона була використана сигмоїда, задача якої — звуження діапазону вихідних значень в межах між 0 та 1.

При навчанні функцією втрат слугувала бінарна перехресна ентропія. Для оптимізації параметрів моделі використовувався алгоритм Adam (адаптивна оцінка моменту).

Для більш достовірного порівняння отриманих моделей класифікатор перетреновувався десять разів (до та після доповнення), після цього для кожного випадку обиралась модель з найвищими показником AUROC (площа під кривою робочої характеристики приймача) на тестувальних даних.

Після отримання базової моделі з достатньо високими показниками ефективності на тестовому наборі даних (AUROC = 0,932, точність = 0,88, при порозі вірогідності = 0,51) було проаналізовано її вразливість до змагальних атак. Для цього для кожної точки в просторі тестових даних було підраховано вектори градієнтів моделі у просторі ознак. Далі, методом, описаним у попередніх розділах, було отримано змагальні приклади для кожної точки базового набору через масштабування значень у напрямку градієнта, окремо для кожного зазначеного набору ознак.

Для кожного значення ε підраховано відсоток міскласифікації екземплярів, отриманих описаним способом, з оригінальних мережевих потоків, які належали класу «ботнетів». Результати наведені в табл. 2. Як зауважувалося раніше, на етапі, коли для конкретного ε трансформація стає недійсною (порушуються встановлені обмеження) для певного потоку, він та його оригінал виключається з експерименту. Відповідно для більш коректної інтерпретації результатів також необхідно слідкувати за загальною кількістю потоків, які розглядаються на даному етапі.

Нижче наведено псевдокод, який показує порядок кроків генерації змагальних прикладів та оцінки вразливості до них моделі.

INPUT
F : ознаки
FG : група ознак, над якою проводяться збурення
C : обмеження
X : вектори ознак мережевих потоків
Y : мітки мережевих потоків (0 — звичайний, 1 — злоякісний/ботнет)
E : величина збурення ({0; 0,01; 0,05; 0,1; 0,2; 0,3; 0,4; 0,5; 0,6; 0,7})
t : поріг впевненості
BEGIN

$f \leftarrow \{1 \text{ if } f_i \in FG \text{ else } 0\}_1^{\|F\|}$ // вектор-селектор групи ознак

$X_{\min}, X_{\max} \leftarrow \{\min_{\forall x \in X} x_i\}_1^{\|F\|}, \{\max_{\forall x \in X} x_i\}_1^{\|F\|}$ // визначити границі ознак

$\theta \leftarrow train(\theta, J(scale(X, X_{\min}, X_{\max}), Y))$ // навчання моделі

// θ — параметри моделі, *J* — функція втрат (бінарна перехресна ентропія)

$X_{adversarial} \leftarrow \{\}$

foreach ε in *E*:

foreach x^n in *X*:

$\overrightarrow{\nabla_{x^n}} \leftarrow \nabla_{x^n} J(\theta, x^n, y^n) * f$ // підрахувати градієнт

$x^n_{adversarial} \leftarrow x^n * (1 + \varepsilon * sign(\overrightarrow{\nabla_{x^n}}))$ // внести збурення

end

$X_{adversarial} \leftarrow \{x^n_{adversarial} \mid constraints_{hold}(x^n_{adversarial}, C)\}$

$P \leftarrow predict(\theta, scale(X_{adversariai}, X_{\min}, X_{\max}))$

$X_{adv\_malicious} \leftarrow \{x^n \mid y^n = 1,\ y^n \in Y, x^n \in X_{adversarial}\}$

$X_{adv\_misclassified} \leftarrow \{x^n \mid p^n > t,\ p^n \in P, x^n \in X_{adv\_malicious}\}$

$misclassification\_rate[e] \leftarrow \frac{X_{adv\_misclassified}}{X_{adv\_malicious}}$

end

END

Зауважим, що для більшої наглядності в табл. 2 наведено не абсолютну величину відсотка міскласифікації, а її приріст відносно початкового набору даних без збурень.

Таблиця 2

	Група ознак 1		Група ознак 2	
ε	**Приріст міскласифікації, %**	**Кількість розглянутих екземплярів**	**Приріст міскласифікації, %**	**Кількість розглянутих екземплярів**
0	0	70024	0	70024
0,01	0,01	69663	0,07	67431
0,05	0,07	69658	0,80	67388
0,1	0,34	69654	2,6	67040
0,2	0,61	69645	4,52	66535
0,3	0,83	69635	9,12	66521
0,4	0,99	69450	11,35	66506
0,5	1,08	69286	12,44	66491
0,6	1,40	69263	13,00	66464
0,7	1,48	69240	13,21	66457

Як видно з табл. 2, модель маловразлива (хоча й не повністю стійка) до змагальних атак описаним методом — варіювання міжпакетних інтервалів та тривалості потоку (група ознак 1). Водночас маніпуляції статистиками розмірів пакетів (група ознак 2) вплинули на модель значно сильніше.

Побудова доповненого навчального набору

Після виявлення вразливості базової моделі до змагальних атак постає задача побудови нового навчального набору, розширеного змагальними прикладами, і подальшого навчання на ньому нової моделі класифікації. При цьому можна виділити два основні критерії, які отримана модель має задовільняти:

1) нова модель класифікації має бути більш стійка до змагальних атак (принаймні до класу атак, представленого у навчальному наборі);

2) загальні показники якості отриманої моделі, такі як точність та відсоток хибно-позитивних результатів, мають бути не гірші, ніж у базової.

Слід зауважити, що в ході виконання роботи зроблено кілька спроб доповнення навчального набору змагальними прикладами, створеними наведеним вище методом, для обох вказаних груп ознак, використовуючи різні значення ε. Доповнення навчальної вибірки прикладами, отриманими для ознак групи 1, як видно з табл. 2, не принесло значних результатів. Базова модель слабо чутлива до варіацій в міжпакетних інтервалах. Однак при доповненні екземплярами, згенерованими для ознак групи 2 (статистики розмірів пакетів), стійкість до змагальних атак значно покращилась, і модель задовільнила обидва критерії.

Найкращий результат принесло розширення навчального набору екземплярами, отриманими з початкових масштабувань у напрямку градієнта ознак групи 2 для ε = [0,05; 0,1; 0,2]. Утворена навчальна вибірка описана в табл. 1.

Результуюча модель, яка навчалась на доповненому наборі, більш стійка до варіювання розмірів пакетів описаним методом порівняно з оригінальною (рис. 2). Додатковим стороннім ефектом стало й те, що хоча доповнення і не включало в себе збурення ознак групи 1, однак на діапазоні ε від 0,1 до 0,3 вразливість до змагальних атак такого роду над цією групою також дещо покращилась (рис. 3). Таким чином, навчена модель задовільняє критерій 1.

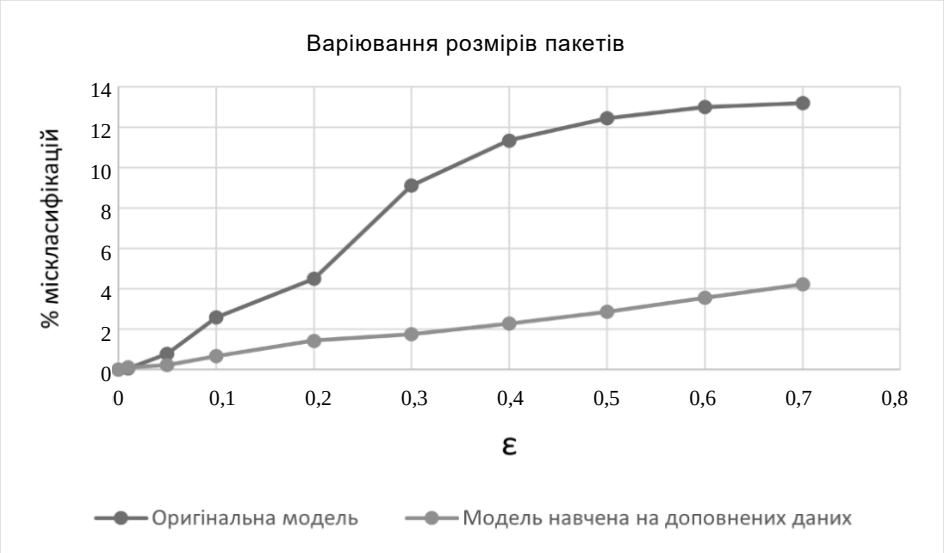

Рис. 2

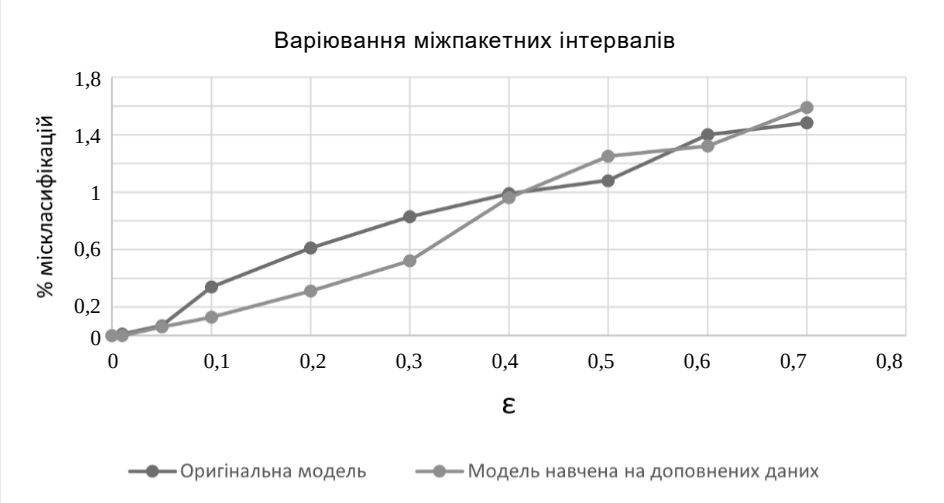

Рис. 3

Попри виконання критерію підвищеної стійкості до змагальних атак, необхідно пересвідчитись, що таке штучне розширення навчального набору не погіршує якість класифікатора за іншими параметрами ефективності. В табл. 3 наводиться порівняльна характеристика вирахуваних показників базової моделі та моделі, навченої на розширених даних. Для обох моделей підраховано показники ефективності.

1. AUROC (площа під ROC-кривою):

$$\text{AUROC} = \int_0^1 TPR(t)FPR'(t)\partial t, \tag{3}$$

де $TPR(t)$ — функція частки істиннопозитивних результатів, $FPR(t)$ — функція частки хибнопозитивних результатів залежно від порогу впевненості t.

2. Точність:

$$Accuracy = \frac{TP + TN}{N}. \tag{4}$$

Точність вираховується за формулою (4), де *TP* — кількість істиннопозитивних результатів, *TN* — кількість істиннонегативних результатів, підрахованих для фіксованого порогу, а *N* — загальна кількість прикладів.

Частка помилок I та II роду:

$$FPR = \frac{FP}{N}, \tag{5}$$

$$FNR = \frac{FN}{N}. \tag{6}$$

Частка помилок I роду вираховується за формулою (5), де *FP* — кількість хибнопозитивних результатів. Частка помилок II роду вираховується за формулою (6), де *FN* — кількість хибнонегативних результатів, при обраному порозі впевненості.

Зауважимо, що під «порогом впевненості» мається на увазі величина $t \in [0, ..., 1]$, з якою порівнюється значення впевеності, передбачуване моделлю для кожного елемента вхідної вибірки. Гіпотеза класифікації приймається для даного елемента, якщо значення, спрогнозоване моделлю, перевищує поріг *t*, а інакше — відкидається. У даній задачі гіпотеза класифікації — це належність мережевого потоку до трафіку ботнетів:

$$\hat{y}_i = \begin{cases} +, \text{ якщо } p(x_i) > t, \\ - \qquad \text{інакше.} \end{cases} \tag{7}$$

Логіка прийняття чи відкидання гіпотези класифікації залежно від порогу впевненості наведена в (7), де $p(x_i)$ — це впевненість класифікатора щодо екземпляра x_i.

Кожному елементу x_i класифікатор ставить у відповідність мітку $\hat{y}_i$. Після цього вона порівнюється з «фундаментальною істиною», відомою з початкової розмітки набору даних *i*, таким чином, робиться висновок про істинність чи хибність припущення.

Таблиця 3

Дані	**Навчальні**		**Тестувальні (нові)**	
Модель	**До**	**Після**	**До**	**Після**
AUROC	0,976	0,981	0,932	0,932
Точність (поріг ймовірності 0,51)	0,935	0,937	0,880	0,891
Частка помилок I роду (поріг ймовірності 0,51)	0,032	0,041	0,061	0,053
Частка помилок II роду (поріг ймовірності 0,51)	0,157	0,122	0,204	0,190

Висновок

У даній роботі описано та реалізовано спосіб аналізу вразливості систем виявлення вторгнень до змагальних атака на базі мережевих потоків та нейронних мереж класифікаторів.

Порівняно з попередніми результатами досліджень у даній області [15, 16], які приймають «як є» методи генерації змагальних прикладів з області обробки зображень, запропонований метод створення змагальних прикладів мережевих потоків адаптований до низьковимірного та гетерогенного простору ознак. Крім того, хоча в деяких роботах [17, 18] при генерації нових значень враховуються статистичні рамки окремих ознак, однак не вводяться жодні міжознакові логічні обмеження, що також враховано в даній публікації.

Ще однією особливістю даного методу є враховування семантики ознак потоків: виділення логічних груп статистик і побудова змагальних даних окремо, супроти кожної групи перевіряється вразливість моделі та підвищення її стійкості. На відміну від інших робіт, де ціллю є аналіз і покращення універсальної стійкості моделі до змагальних атак на всьому просторі ознак (не залежно від того, як і чи взагалі можливо реалізувати таку атаку на практиці), запропонований метод дозволяє робити більш конкретні судження: вразливість/стійкість до «варіації міжпакетних інтервалів» чи «маніпуляції з розміром корисного навантаження пакетів». Таке семантичне тлумачення схвалюється кібербезпекою і може використовуватися для формулювання конкретних вимог і гарантій до системи виявлення загроз.

Також експериментально продемонстровано ефективність доповнення навчальних даних змагальними прикладами, отриманими наведеним методом, що забезпечило помітне підвищення стійкості моделі до атак представленого типу, не погіршуючи практичні генералізуючі властивості класифікатора.

Описаний підхід є основою для подальших досліджень, зокрема, в напрямках більш якісної генерації змагальних прикладів не лише однокроковими, а й ітеративними методами, а також отримання більш стійких моделей методами змагального навчання.

Додаток

Наведено список з 50 ознак, отриманих інструментом CICFlowMeter [14], яким було представлено мережеві потоки, що використовувалися в експериментальній частині даної роботи.

Номер	Оригінальні назви ознак	Тип даних	Опис
1	Flow Duration	int	Тривалість потоку
2–3	Total Fwd/Bwd Packet	int	Кількість пакетів у потоці
4–5	Total Length of Fwd/Bwd Packet	int	Сумарна довжина пакетів у потоці
6–11	Fwd/Bwd Packet Length Min/Max/Std	float	Мінімальна/Максимальна/Середньоквадратичне відхилення довжини пакета
12–19	Fwd/Bwd IAT Total/Min/Max/Std	float	Сумарне/Мінімальний/Максимальний/Середньоквадратичне відхилення інтервалу між пакетами у потоці
20–23	Fwd/Bwd PSH/URG Flags	int	Кількість пакетів з встановленими TCP-прапорцями PSH/URG
24–25	Fwd/Bwd Header Length	int	Сумарна довжина заголовків пакетів у потоці
26–33	FIN/SYN/RST/PSH/ACK/URG/CWR/ECE Flag Count	int	Кількість пакетів з встановленими TCP-прапорцями FIN/SYN/RST/PSH/ACK/URG/CWR/ECE
34	Down/Up Ratio	float	Відношення кількості завантажень до вивантажень
35–36	Fwd/Bwd Segment Size Avg	float	Середній розмір сегмента у потоці
37–40	Fwd/Bwd (Bytes/Packet)/Bulk Avg	int	Середня кількість байтів/пакетів у кластері пакетів
41–42	Fwd/Bwd Bulk Rate Avg	int	Середня кількість байтів на секунду в кластерах пакетів потоку
43–46	Subflow Fwd/Bwd Packets/Bytes	int	Кількість пакетів/байтів у підпотоці
47–48	Fwd/Bwd Init Win Bytes	int	Кількість байтів, передана в початковому вікні
49	Fwd Act Data Pkts	int	Кількість пакетів хоча б з одним байтом даних, переданих від джерела до приймача
50	Fwd Seg Size Min	int	Мінімальна довжина сегмента у потоці

GENERATION AND USE OF ADVERSARIAL SAMPLES TO COUNTER BOTNET EVASION FROM NEAURAL NETWORK DETECTION

(to the 100th anniversary of academician V.M. Glushkov birthday)

Bohdan Panchuk

V.M. Glushkov Institute of Cybernetics of NAS of Ukraine, Kyiv,

bogdanscloud@gmail.com

In this paper, we describe a method for assessing the reliability of botnet detection systems based on neural networks in terms of their susceptibility to adversarial attacks and enhancing the resilience of such systems through the augmentation of the training dataset with synthetic data. By «adversarial attack», we mean a deliberate attempt by an attacker to modify malicious data (in our case, network packet streams) in an effort to evade being detected by the classifier. In scope of this work, we implemented a method for generating adversarial examples for a traffic classification system based on a neural network, adapting the «fast sign gradient method» commonly used in image processing to work with network data represented as network flows. Key attributes of the suggested approach are computational simplicity and the plausibility of the generated traffic examples. Plausibility was ensured by imposing global and inter-feature constraints and identifying semantically related feature groups subjected to the modifications. In addition to applying the described approach to assess the vulnerability of classification models, we also demonstrated its applicability for augmenting the 19 initial training dataset with synthetic data. Initially, we trained the baseline classification model on an open dataset of botnet traffic. Subsequently, we augmented the initial dataset with adversarial examples generated using the described method. Experimental results showed that the model trained on the augmented data exhibited greater resilience to adversarial attacks in comparison to the baseline model. Importantly, this method is not specific to botnet detection and can be applied to other types of network attacks given an appropriate training dataset. In conclusion, we summarized our findings and suggested directions for further development of this approach.

Keywords: network flows, neural networks, intrusion detection systems, adversarial attacks, data augmentation, botnets.

ПОСИЛАННЯ

1. Karve Swagat, Arpityadav, Dutta Prateek. Artificial intelligence in cyber security. *REST Journal on Emerging trends in Modelling and Manufacturing.* 2022. N 8. DOI: https://dx.doi.org/10.46632/jemm/8/2/6
2. Thanh Vu, Simon Nam, Mads Stege, Peter Issam El-Habr, Jesper Bang, and Nicola Dragoni. A survey on botnets: incentives, evolution, detection and current trends. *Future Internet.* 2021. Vol. 13, N 8. 198 p. DOI: https://doi.org/10.3390/fi13080198
3. Szegedy C., Zaremba W., Sutskever I. et al. Intriguing properties of neural networks. 2013. DOI: https://doi.org/10.48550/arXiv.1312.6199
4. Geman S., Bienenstock E., Doursat R. Neural networks and the bias/Variance Dilemma. *Neural Computation*. 1992. N 4. P. 1–58. DOI: https://doi.org/10.1162/neco.1992.4.1.1
5. Goodfellow Ian J., Shlens Jonathon, Szegedy Christian. Explaining and harnessing adversarial examples. 2014. DOI: https://doi.org/10.48550/arXiv.1412.6572

6. Letteri I., Rosso M.D., Caianiello P., Cassioli D. Performance of botnet detection by neural networks in software-defined networks. *Italian Conference on Cybersecurity*. 2018. URL: https://ceur-ws.org/Vol-2058/paper-03.pdf
7. Arnaldo Ignacio, Cuesta-Infante Alfredo, Arun Ankit, Lam Mei, Bassias Costas, Veeramachaneni Kalyan. Learning representations for log data in cybersecurity. 2017. P. 250–268. DOI: https://dx.doi.org/10.1007/978-3-319-60080-2_19
8. Meshal Farhan AL-Anazi and Mostafa G M Mostafa. Efficient botnet detection using feature ranking and hyperparameter tuning. *International Journal of Computer Applications*. 2019. 182, N 48. P. 55–60. DOI: https://dx.doi.org/10.5120/ijca2019918739
9. Beigi E.B., Jazi H.H., Stakhanova N., Ghorbani A.A.: Towards effective feature selection in machine learning–based botnet detection approaches. *IEEE Conference on Communications and Network Security*. 2014. P. 247–255. DOI: https://dx.doi.org/ 10.1109/CNS.2014.6997492
10. Панчук Б.О. Виявлення ботнет-трафіку на основі мережевих потоків методами ШІ. *Проблеми програмування.* 2022. № 3-4. С. 376–386. DOI: https://doi.org/10.15407/pp2022.03-04.376
11. Rumelhart D.E., Hinton G.E., Williams R.J. Learning representations by back-propagating errors. *Nature*. 1986. N 323. P. 533–536. DOI: https://doi.org/10.1038/323533a0
12. Terence Parr, Jeremy Howard. The matrix calculus you need for deep learning. 2018. DOI: https://doi.org/10.48550/arXiv.1802.01528
13. CICFlowMeter (formerly ISCXFlowMeter) URL: https://www.unb.ca/cic/research/applications.html
14. AsSadhan Basil, Bashaiwth Abdulmuneem, Al-Muhtadi Jalal, Alshebeili Saleh. Analysis of P2P, IRC and HTTP traffic for botnets detection. *Peer-to-Peer Networking and Applications.* 2018. DOI: https://link.springer.com/article/10.1007/s12083-017-0586-0
15. Zhang Xingwei, Xiaolong Zheng, Wu Desheng. Attacking DNN-based intrusion detection models. *IFAC-PapersOnLine.* 2020. N 53. P. 415–419. DOI: https://doi.org/ 10.1016/j.ifacol.2021.04.118
16. Hu Yongjin, Tian Jin, Ma Jun. A novel way to generate adversarial network traffic samples against network traffic classification. *Wireless Communications and Mobile Computing*. 2021. P. 1–12. DOI: https://dx.doi.org/10.1155/2021/7367107
17. Zolbayar Bolor, Sheatsley Ryan, McDaniel Patrick, Weisman Michael, Zhu Sencun, Zhu Shitong, Krishnamurthy Srikanth. Generating practical adversarial network traffic flows using NIDSGAN. 2022. DOI: https://doi.org/10.48550/arXiv.2203.06694
18. Randhawa Rizwan, Aslam Nauman, Alauthman Mohammad, Khalid Muhammad, Rafiq Husnain. Deep reinforcement learning based evasion generative adversarial network for botnet detection. 2022. DOI: https://doi.org/10.48550/arXiv.2210.02840

Отримано 26.07.2023