УДК 004.8, 519.7

О.О. ЛЕТИЧЕВСЬКИЙ
Інститут кібернетики ім. В.М. Глушкова НАН України, Київ, Україна,
e-mail: *oleksandr.letychevskyi@litsoft.com.ua*.

КОГНІТИВНІ МЕРЕЖІ, ЇХНІ ВЛАСТИВОСТІ ТА ЗАСТОСУВАННЯ У СИСТЕМАХ ВИЯВЛЕННЯ ТА ЗАПОБІГАННЯ АТАКАМ

Анотація. Розглянуто методи виявлення кібератак у реальному часі, що ґрунтуються на алгебраїчному підході. Застосовано метод алгебраїчного зіставлення, в основу якого покладено методи розв'язання поведінкових рівнянь та алгебраїчного моделювання. Водночас для підвищення ефективності виявлення та запобігання запропоновано використовувати композицію нейронної мережі для класифікації атак та методу алгебраїчного зіставлення. Цю конструкцію називають когнітивною мережею, поняття якої вперше було сформульовано у 2005 р. Розглянуто такі властивості когнітивної мережі, як подвійна класифікація та самонавчання. Представлено технологічну лінію для виявлення кібератак з використанням когнітивних мереж.

Ключові слова: штучний інтелект, нейронна мережа, глибоке машинне навчання, алгебра поведінок, нейросимвольний підхід, алгебраїчне моделювання, інсерційне моделювання.

ВСТУП. ПРОБЛЕМИ МЕТОДІВ ШТУЧНОГО ІНТЕЛЕКТУ У ВИЯВЛЕННІ КІБЕРАТАК

Методи штучного інтелекту сьогодні досить активно застосовують у різних предметних галузях. Зокрема технологію нейронних мереж використовують у таких задачах класифікації, як розпізнання образів, розуміння мови, у галузі кібербезпеки, медицини та у багатьох інших. Із моделями класифікації співіснують і так звані генеративні мережі, що створюють такі об'єкти, як зображення, мовні конструкції тощо, згідно з навчанням нейронної мережі. Зокрема, в галузі кібербезпеки генеративні системи використовують для створення наборів даних з певними властивостями, на яких навчають нейронну мережу.

Застосування технології штучного інтелекту для виявлення у реальному часі атак на мережеве або комп'ютерне середовище є однією із нагальних задач кібербезпеки, з якою пов'язані численні розробки, як інженерні, так і наукові. Задача полягає як у виявленні ознак вторгнення, так і в нейтралізації атаки у межах комп'ютерних ресурсів. Для захисту від цих зловмисних дій аналізують як дані протоколів інтернету, так і поведінку системи, та у разі виявлення активують захисні дії для запобігання атаці.

У сучасній індустрії кібербезпеки використовують системи виявлення вторгнень, які є удосконаленням попередніх функцій захисту, як-от: мережеві екрани, віртуальні приватні мережі та інші засоби. Для швидкого виявлення атаки застосовують нейронні мережі глибокого навчання DNN (Deep Neuron Networks), які можуть класифікувати поведінку мережевого протоколу під час вторгнення як ненормальну. Більш розвинені нейронні мережі визначають тип атаки відповідно до її класифікаційної моделі.

Нейронні мережі будують шляхом навчання на певних наборах даних, зібраних під час спостереження за поведінкою мережевих протоколів. Щоб у найпростіший і найшвидший спосіб виявити атаки в режимі реального часу, потрібно детектувати аномальну поведінку, яка не відповідає звичайним діям протоколу. Але водночас можуть виникнути помилкові виявлення, оскільки відхилення від нормальної поведінки можуть бути спричинені не тільки вторгненням у систему, а й неправильним використанням ресурсів, помилками користувача або програмами, що працюють у середовищі. З іншого боку, досить важко

створити набір даних, який охопить усі сценарії нормальної поведінки хоча б з точністю до класів еквівалентності, а у багатьох випадках — неможливо. Тому перевагу віддають тим системам, які можуть класифікувати атаку або причини аномалії.

Важливу роль відіграють навчальні набори даних для нейронної мережі. Це вхідні дані, на основі яких генерують та налаштовують нейронну мережу.

У відкритому доступі є певна кількість цих наборів. Їх можна використовувати для навчального створення нейронних мереж, але вони досить недосконалі. З наборами, які застосовують для індустріальних сучасних систем, також пов'язані певні проблеми, зумовлені навчанням мережі. Це, по-перше, наявність певного дисбалансу між класифікаціями різних атак — для одних атак є більша кількість даних, для інших — менша. З іншого боку, навчена нейронна мережа забезпечуватиме високу точність класифікації лише в тих умовах, в яких вона була навчена.

Сучасні хакери розуміють природу моделі класифікації атак і намагаються замаскувати свої дії під звичайну поведінку у протоколі або ухилитись від класифікації. Ці дії визначають як змагальну атаку. У зв'язку з ними виникає проблема подолання такого ухилення (evasion) хакерів.

Під час протидії системам виявлення вторгнень хакер використовує еквівалентні перетворення у своїх діях, щоб вони не були розпізнані нейронною мережею як атака. Такими перетвореннями можуть бути перестановка компонентів атаки, вставка додаткових символів-роздільників, використання альтернативних кодувань, нестандартних портів і безліч інших хитрощів, які не дають змоги класифікувати поведінку мережевого протоколу як такого, що містить атаку.

Одним зі способів розв'язання проблеми змагальних атак є створення доповненого набору даних, який застосовує можливі відомі перетворення, використовувані хакерами. Доповнення цим набором робить систему більш надійною та забезпечує можливість виявлення вторгнень з можливими ухиленнями від класифікації.

АЛГЕБРАЇЧНИЙ ПІДХІД У КІБЕРБЕЗПЕЦІ

Застосування формальних методів уможливило реалізацію більш надійного способу запобігання атакам. Уперше відповідне підтвердження отримано у 2016 р. під час змагання з виявлення вразливостей та протидії кібератакам, проведеного агенцією DARPA [1]. Учасники, які посіли три перших місця, використовували лише алгебраїчний підхід.

Основний метод запобігання атакам — це виявлення вразливостей у системах з використанням формальних методів. Для виявлення атак у реальному часі застосовують методи RunTime Verification (верифікаціїї у процесі виконання). Під час цієї верифікації здійснюють моніторинг всіх процесів у мережевому середовищі та перевіряють їх на властивості безпеки. Зокрема цей метод застосовують для об'єктів критичної інфраструктури, де створюють формальну модель системи, на якій здійснюють верифікацію на властивості безпеки під час роботи системи. У роботі [2] розроблено систему розподілу води, де зберігається її модель у вигляді часових автоматів з використанням відповідної формальної верифікації.

В.М. Глушков створив потужну алгебраїчну школу в Інституті кібернетики НАН України. Завдяки цьому та здобуткам у розвитку методів автоматичного доведення теорем склалися передумови для виникнення алгебраїчного підходу до розв'язання практичних задач. Так було створено систему алгебраїчного програмування [3] та теорію взаємодії агентів і середовищ [4] (спільно з британським науковцем Д. Гільбертом). Результатом подальшого розвитку цих досліджень за участю автора було створення теорії інсерційного моделювання [5] та алгебри по-

ведінок [6], які є узагальненням теорій автоматів та транзиційних систем (transition system). Алгебра поведінок виявилася досить розвинутим математичним апаратом для формалізації як математичних, так і природних об'єктів. На основі алгебри поведінок було розроблено низку формальних методів.

Відому поведінку зловмисника можна формально описати за допомогою поведінкових рівнянь, технологію розв'язання яких розробили О. Летичевський та В. Песчаненко [7]. Для цього використовують основні поняття теорії агентів і середовищ.

Агент — це сутність яка виконує дії, що змінюють стан середовища. Поняття «агент» є еквівалентним поняттю «автомат» або «транзиційна система». Множина послідовностей поведінок або сценаріїв формує поведінку агента. Стан агента визначається множиною його типізованих атрибутів. Агенти взаємодіють у деякому середовищі, обмінюючись сповіщеннями, або через відповідні атрибути середовища. Розглядають багаторівневу систему агентів та середовищ, тобто агент може бути середовищем, в якому взаємодіють агенти нижчого рівня абстракції. У свою чергу середовище може бути агентом, який взаємодіє в межах середовища вищого рівня абстракції. Теорія агентів та середовищ визначає функцію занурення агента в середовище, а саме появу агента у середовищі, взаємодію його з іншими агентами та видалення із середовища. Функцію занурення представляють за допомогою рівнянь алгебри поведінок.

Агент має тип, що визначається множиною його атрибутів. Атрибути агента є типізованими, наприклад, цілочисловими, символьними, бітовими, булевими. Кожен тип представляє певну теорію із множиною операцій та предикатів.

Алгебра поведінок — це двосортна універсальна алгебра. Основний сорт — це множина поведінок агентів, а другий сорт — це множина дій. Ця алгебра має дві операції: префіксінг $a.u$ (де a — дія, а u — поведінка) та альтернативний вибір $u+v$ (u та v — поведінки). Алгебра поведінок також збагачена двома операціями: паралельною $(u||v)$ і послідовною $(u;v)$ композиціями поведінок. В алгебрі поведінок використовують такі термінальні константи: успішне завершення поведінки Δ, тупиковий стан 0 та невизначена поведінка $\perp$.

Семантику дії кожного агента визначають як трійку $B=\langle P, A, Q\rangle$, де P — передумова дії, представлена у вигляді формули над атрибутами агента, Q — постумова, а A — процес, який визначає ілюстрацію дії агента під час переходу між станами. Загалом семантика дії означає, що агент може змінити свій стан, якщо передумова істинна, а стан зміниться відповідно до постумови. Постумова може містити оператор присвоєння. Процес дії залежить від предметної галузі та ілюструє послідовність застосування дій. У домені протоколу зв'язку це може бути надсилання або отримання сигналів із відповідними параметрами.

Поведінковий вираз — це терм у сигнатурі алгебри поведінок, який визначається послідовністю дій, поведінки та операціями алгебри поведінок.

Поведінкову систему рівнянь задають набором рівностей. У лівій частині знаходиться унікальне ім'я змінної, що визначає поведінку, а у правій — вираз поведінки:

$$B_{11}=T_1(a_{i11}, a_{i12}, \ldots, B_{i11}, B_{i12}, \ldots),$$
$$B_{12}=T_2(a_{i21}, a_{i22}, \ldots, B_{i21}, B_{i22}, \ldots)\ldots$$

Початкова поведінка у системі рівнянь поведінки завжди визначена. Систему рівнянь поведінки можна перевести в канонічну нормальну форму:

$$Bi=\sum_j a_{ij}.B_{k_{ij}}+\varepsilon, \text{ де } \varepsilon\in\{\Delta, \perp\}.$$

Підстановкою $R(B, B_x)$ поведінки B_x у рівняння поведінки B, яка містить змінну B_x у правій частині, є перетворення, в якому замість змінної поведінки B_x виконують підстановку рівняння $B_x=(\ldots)$.

Розгортанням системи поведінкових рівнянь $B_0 = G(B_i)$ є набір підстановок $R(B_0, B_{i1}), R(B_{i1}, B_{i2}), \ldots$, де $B_{i1}, B_{i2}, \ldots$ — змінні поведінки, що містяться у правій частині B_{ij}, або з'являються у правій частині в результаті замін. Результатом повного розгортання рівняння поведінки є послідовність дій, яку називають трасою. Траса — це послідовність дій, яка відповідає поведінковому рівнянню, або поведінка, яка була повністю розгорнута.

Якщо розглядати поведінку без урахування семантики дій, то можна говорити про розв'язання системи рівнянь поведінки. Тут розглянуто модель поведінки B з довільними поведінками $X_1, X_2, \ldots$, поведінковими змінними $B_1, B_2, \ldots$ та діями $a_1, a_2, \ldots$:

$$B = G(X_1, X_2, \ldots, B_1, B_2, \ldots, a_1, a_2).$$

Також маємо систему рівнянь поведінки B_0. Потрібно виявити досяжність поведінки B у системі рівнянь поведінки B_0. Розв'язком системи поведінкових рівнянь B_0 відносно B є множина трас, які задовольняють такі умови:

- їх можна отримати з B_0 шляхом розгортання;
- вони повинні бути зіставлені з B.

Метод розв'язання систем рівнянь поведінки передбачає перетворення системи до канонічної нормальної форми та її відповідне розгортання згідно з шуканою поведінкою.

Нехай початковий стан агентів і середовища представлено початковою формулою S_0. Починаючи з початкової формули, можна застосувати дію з подальшим розгортанням рівнянь алгебри поведінки. Дія є застосовною, якщо її передумова є виконуваною та сумісною з поточним станом. Виходячи з формули початкового стану S_0 та початкової поведінки B_0, вибирають застосовну дію та переходять до наступної поведінки. Для цього на першому кроці перевіряють виконуваність кон'юнкції $S_0 \wedge P_{a_1}$, якщо $B_0 = a_1.B_1$, а P_{a_1} є передумовою a_1. Наступний стан середовища отримують за допомогою предикатного перетворювача [8], тобто функції над поточним станом агента, передумовою та постумовою:

$$PT(S_i, P_{a_i}, Q_{a_i}) = S_{i+1}.$$

Шляхом застосування функції предикатного перетворювача до різних станів агента можна отримати послідовність $S_0, S_1, \ldots$ формул, які виражають стани агента, що змінюються від початкового стану. Трасу можна представити послідовністю процесів дій $a_1, a_2, \ldots$ Кожен стан агента охоплює множину конкретних значень атрибутів агента, а процес генерації цих трас називають алгебраїчним моделюванням. Для кожної теорії, в якій проводять алгебраїчне моделювання, потрібно визначити відповідний предикатний перетворювач.

Алгебраїчне зіставлення — це метод пошуку поведінки, яка відповідає шаблону в системі поведінкових рівнянь. Шаблон можна також представити як систему поведінкових рівнянь, а набір дій шаблону можна доповнити додатковими умовами у перед- і постумовах. Шаблон може містити невідомі (довільні) частини поведінки.

Метод алгебраїчного зіставлення містить два кроки:

— знайти послідовність дій, яка відповідає шаблону;

— довести досяжність поведінки (послідовності дій) з початкового стану.

На першому кроці розв'язують рівняння поведінки відносно поведінки, представленої шаблоном.

На другому кроці виконують алгебраїчне моделювання від початкової формули до останньої дії поведінки.

У галузі кібербезпеки, якщо шаблон описує можливу атаку, а система задана поведінковим рівнянням, то розв'язком рівняння є послідовність дій, що призведе до цієї атаки. Це використовують у системі виявлення вторгнень у реальному часі.

За допомогою методу алгебраїчного зіставлення атаку можна виявити шляхом моніторінгу мережевого протоколу та використання моделі системи чи сегмента мережевого оточення з метою запобігання вторгненню. Для цього застосовують поведінкові шаблони атак.

Як приклад, шаблон атаки пошкодження стеку можна представити такою системою поведінкових рівнянь:

StackDamage = syscall(1,null,null).Z,
Z=(X1 + Cnt;Z) + _call(2,Numeric,null).{Y},
Cnt =mov(4,Memory(x,y,z),regm)+mov(5,Memory(x,y,z),regm)+mov(6,regm,
Memory(x,y,z))+mov(7,regm, Memory(x,y,z))+mov(8,regm, regn)+mov(9,regm, regn),
Y=(Cnt+X2;Y)+mov(3,Memory(ds,y,z),regn).Delta.

Ця атака має на меті руйнування стеку програми або перезапис коду повернення у вершині стеку з метою перехоплення керування.

Для такої атаки поведінковий шаблон створюють для бінарного коду системи, точніше для множини машинних інструкцій процесора Intel x86. Семантику кожної інструкції представляють як дію в алгебрі поведінок, що має відповідні перед- та післяумови. Поведінкове рівняння визначатиме послідовність дій у шаблоні. Вона може бути як лінійною послідовністю, так і деревом. Поведінкове рівняння може містити ідентифікатори довільних поведінок.

Сам шаблон створюють як послідовність критичних точок, починаючи від уведення деяких даних із різних точок входу, зокрема із зовнішнього середовища, до можливого результату атаки. У цьому разі пошкодження стеку розглядають як кінцевий результат атаки або експлуатацію вразливості.

Для виявлення атаки в бінарному коді програмної системи потрібно мати модель цієї системи в алгебрі поведінок. Якщо шаблони атак створюють вручну, то модель системи створює відповідний транслятор, який генерує відповідну систему поведінкових рівнянь та множину дій в автоматичному режимі.

На рис. 1 представлено приклад цієї трансляції.

Задача виявлення атаки в реальному часі зводиться до алгебраїчного зіставлення поведінки системи після отримання даних із зовнішнього середовища із шаблоном атаки.

З одного боку, якщо система сприймає всі вхідні дані, то атака буде виявлена, коли вторгнення вже відбудеться. Тому запропоновано організувати роботу системи виявлення, як мережевий екран, який здійснює фільтрацію вхідних даних, визначених як шкідливі.

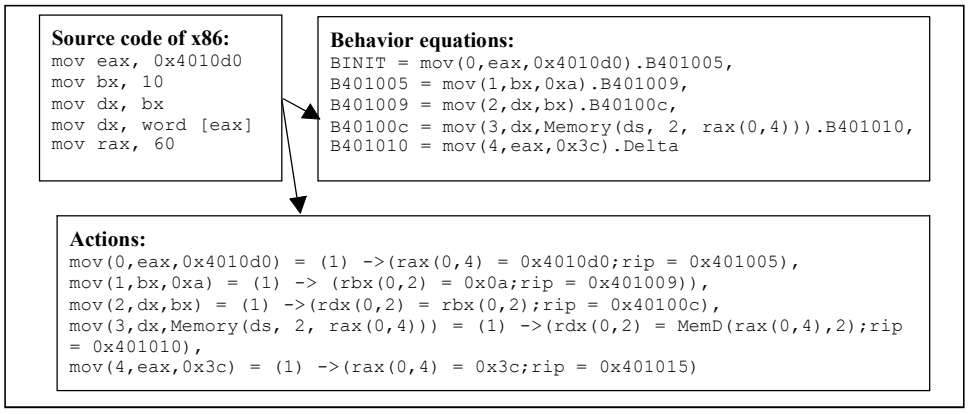

Рис. 1. Трансляція машинних інструкцій асемблера Intel x86 у специфікації алгебри поведінок

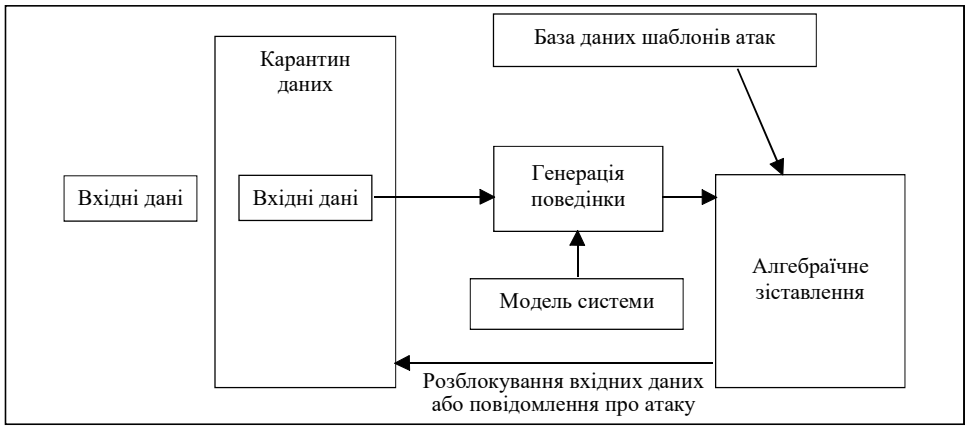

Рис. 2. Схема виявлення атак методом алгебраїчного зіставлення

Розглянемо схему методу виявлення атаки та запобігання їй, наведену на рис. 2.

Система виявлення вторгнень та запобігання їм сприймає вхідні дані із зовнішнього середовища та поміщає їх в карантин. Водночас ініціалізують поведінкову модель тієї системи, на яку може бути націлена атака. За допомогою моделювання генерують відповідну поведінку цієї системи під дією вхідних даних. Далі здійснюють алгебраїчне зіставлення можливих шаблонів атак зі згенерованою поведінкою. У тому разі, коли зіставлення відбулося та згенерована поведінка досягла закінчення атаки, дані блокуються повністю та видається попередження про атаку. Якщо атака не досягла закінчення, то система виявлення приймає наступну порцію даних і процедура знову повторюється. Якщо зіставлення не відбулося, то дані пропускаються в систему.

Іноді дані можуть бути пропущені у систему до визначення атаки, якщо система очікує на відповідь із зовнішнього середовища для подальшого зіставлення із шаблоном.

КОМБІНОВАНИЙ МЕТОД ВИЯВЛЕННЯ АТАК

Алгебраїчний підхід виявився неефективним з погляду швидкості і дані, для яких шукали можливий шаблон атаки, залишалися в карантині протягом деякого проміжку часу, тривалість якого була критичною для нормальної роботи мережі. Нейронна мережа, що працює як модель класифікації, розпізнає атаку набагато швидше, тому варто було використати цю властивість та запропонувати гібрідний метод виявлення атак, що поєднує нейронну та алгебраїчну компоненти.

Якщо побудувати нейронну мережу для тих атак, що були формалізовані як алгебраїчні специфікації, то її можна використати як можливість надання підказок алгебраїчному методу, який визначить точний тип атаки.

Схему технологічної лінії комбінованого методу наведено на рис. 3.

Як і в попередньому випадку, вхідні дані перебувають протягом деякого часу в карантині. Але в цій схемі нейронна мережа сприймає дані раніше та визначає тип атаки. Під час класифікації атаки та визначення її типу інформація передається на алгебраїчну частину в базу даних шаблонів та активується відповідна задача алгебраїчного зіставлення для визначеного шаблону. Це свідчить про те, що ця атака не була хибним виявленням.

Утім, надійність процедури підвищується, коли паралельно працюють обидві компоненти, які підтверджують атаку. У цьому випадку всю інформацію перевірятиме алгебраїчна компонента, але відповідний час аналізу трафіку значно зросте. Хоча, якщо є певність у тому, що нейронна мережа не пропустить атаку, то можна використовувати швидке виявлення.

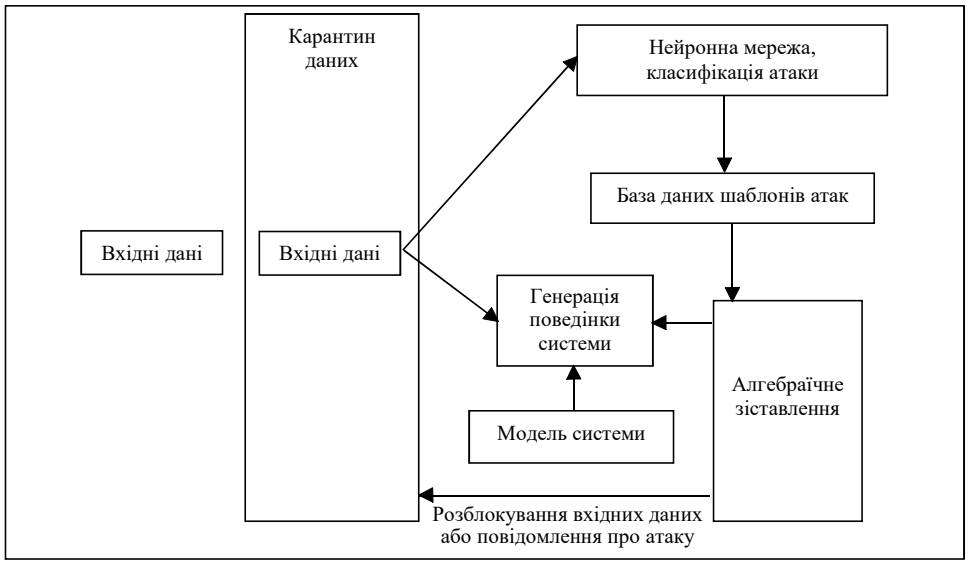

Рис. 3. Схема виявлення атак комбінованим методом

КОГНІТИВНІ МЕРЕЖІ ТА ЇХНІ ВЛАСТИВОСТІ

Отже, маємо композицію двох моделей класифікації. Одна з них є навченою на прикладах нейронною мережею, а друга — алгебраїчною компонентою, що містить модель системи, яка потребує захисту, та базу даних шаблонів.

Цю композицію називають когнітивною мережею. Назва походить від латинського слова cognitio, яке перекладають як «пізнання, вивчення, усвідомлення». Хоча цей термін використовують у декількох контекстах, які значною мірою відрізняються один від одного, проте він завжди означає здатність до розумового сприйняття та перероблення зовнішньої інформації.

У розглянутому тут випадку поєднано дві компоненти. Перша компонента уособлює статистичний досвід, а друга — здатність виводити факти зі знань. Загалом можна визначити це як пізнання на основі прикладів та висновків зі знань разом із прийняттям рішень на основі досвіду та знань.

Цей термін раніше використовували у телекомунікаційній галузі щодо бездротових мереж, в яких була передбачена деяка самоорганізація для більш ефективної роботи відповідних пристроїв, зокрема радіо. Сам термін уперше визначено у [9] як мережу, що має формалізовані знання та може приймати рішення на їхній основі. До того ж, така мережа здатна до самонавчання та модифікації власних знань.

У літературі підхід на основі дедуктивного виведення та навчання за прикладами розглядають як нейросимвольний підхід у штучному інтелекті. Відповідні роботи та огляд наведено в [10].

У цілому, когнітивна мережа — це багатокомпонентна сутність, яка може містити компоненти, що по-різному оперують зі знаннями (машинне навчання, дедуктивне та індуктивне виведення, онтології тощо).

Алгебраїчна компонента є сукупністю поведінкових рівнянь і дій. Розрізняють різні типи когнітивних мереж відповідно до їхніх функцій та властивостей.

Когнітивні мережі-класифікатори. Прикладом когнітивної мережі-класифікатора є розглянутий раніше приклад застосування у виявленні атак: тоді, коли нейронна мережа виявляє підозру на атаку, вона підказує, якого типу атака відбувається. Алгебраїчна компонента здійснює свою класифікацію, результат якої є підтвердженням або спростуванням атаки.

Когнітивні мережі із самонавчанням. Можна активувати когнітивну мережу в режимі, коли працює паралельна композиція її компонент. У цьому режимі мережа здійснює дві класифікації, що мають підтверджувати одна одну. У тому разі, коли навчання нейронної мережі виконано із застосуванням недостатнього набору даних або змінилися умови, в яких відбувалося навчання, класифікації можуть не збігатися. Це також може статися у разі змагальної атаки. Тоді можна застосувати таку важливу властивість когнітивних мереж, як самонавчання в реальних умовах.

Прикладом такого функціонування є класифікація атак обома компонентами. У тому разі, коли результати класифікації не збігаються, застосовують донавчання нейронної мережі з використанням даних, розмічених алгебраїчною компонентою.

Генеративні когнітивні мережі. Когнітивну мережу можна використати для генерації об'єктів на основі знань, які представляють обидві компоненти. Одним із прикладів може бути застосування когнітивних мереж в автоматизації доведень теорем. Алгебраїчна компонента містить поведінкові рівняння, в яких зафіксовані правила виведення, а також аксіоми та доведені теореми в певній теорії. На вхід когнітивної мережі подають теорему, яку потрібно довести. Нейронна мережа навчена на різних формулах і за структурою формули класифікує, яке співвідношення або правило виведення можна застосувати. Алгебраїчна компонента відповідно до підказки виконує наступний крок у доведенні. У такий спосіб відбувається генерація автоматичного доведення з урахуванням підказок нейронної мережі.

ВИСНОВКИ

Когнітивна мережа на основі алгебри поведінок та простого персептрона здатна більш ефективно здійснювати класифікацію атак у мережевому середовищі. Вона працює у двох режимах. У разі застосування більш швидкого режиму в когнітивній моделі класифікації протягом усього основного часу працює нейронна мережа та аналізує дані з карантину з досить високою швидкістю. Алгебраїчна компонента вмикається, коли нейронна мережа виявила підозру на атаку. У цьому випадку вдається практично уникнути False Positive виявлень. Проте якщо нейронна мережа недостатньо навчена, то атака може бути пропущена. Зокрема, залишається загроза змагальної атаки.

Другий, посилений режим, в якому дані сприймаються обома компонентами, дає змогу виявити ті атаки, які точно є атаками, при цьому алгебраїчна компонента класифікації нейронної мережі надає відповідне підтвердження. У цьому режимі також відбувається самонавчання нейронної мережі реальними прикладами із зовнішнього середовища. Коли нейронна мережа буде досить навчена в реальних умовах, то можна здійснити перехід до більш швидкого режиму виявлення. Посилений режим для самонавчання також можна реалізувати на багатопроцесорній системі, як паралельну програму. Паралельну версію алгебраїчної компоненти для виявлення вразливостей реалізовано автором спільно з В. Песчаненком для багатопроцесорного комплексу СКІТ-4 в Інституті кібернетики ім. В.М. Глушкова.

Недоліком технології є тривалий процес формалізації знань. Він полягає у вивченні семантики наявних атак та вразливостей і його здійснюють вручну. Формалізацію шаблонів атак можна виконати швидко, маючи достатні знання, але в ручному режимі можуть бути допущені помилки. Для зменшення ймовірності помилки потрібно застосовувати методи подвійного ревю, тестування шаблонів, залучення експертів.

У межах майбутніх досліджень заплановано приділити увагу автоматизації процесу формалізації, зокрема із застосуванням класичних алгоритмів індуктивного виведення або синтезу скінченних автоматів.

Когнітивні мережі можна застосовувати також тоді, коли про семантику атаки мало що відомо. Шаблони таких атак можуть бути з досить слабкими об-

меженнями. Якщо нейронна мережа навчена досить точно, то додаткова класифікація буде підтвердженням точності виявлення з боку алгебраїчної компоненти. З іншого боку, алгебраїчне зіставлення з шаблонами з більш слабкими обмеженнями працює швидше. Зокрема, за наявності навчального набору когнітивні мережі можна використати для виявлення вразливостей Back Doors, які є штучно створеними прихованими вразливостями з невідомою семантикою.

СПИСОК ЛІТЕРАТУРИ

1. DARPA, "Cyber Grand Challenge." URL: https://www.darpa.mil/program/cyber-grand-challenge.
2. Mercaldo F., Martinelli F., Santone A. Real-time SCADA attack detection by means of formal methods. *Proc. 2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)* (12–14 June 2019, Napoli, Italy). Napoli, 2019. P. 231–236. https://doi.org/10.1109/WETICE.2019.00057.
3. Kapitonova J., Letichevsky A. Algebraic programming in the APS system. *Proc. ISSAC'90: Int'l Symposium on Symbolic Algebraic Computation* (20–24 August 1990, Tokyo, Japan). Tokyo, 1990. P. 68–75. https://doi.org/10.1145/96877.96896.
4. Gilbert D., Letichevsky A. A model for interaction of agents and environments. In: Recent Trends in Algebraic Development Techniques. Bert D., Choppy C. (Eds). *LNCS*. 1999. Vol. 1827. P. 311–328. https://doi.org/10.1007/978-3-540-44616-3_18.
5. Letichevsky A., Letychevskyi O., Peschanenko V. Insertion modeling and its applications. *Computer Science Journal of Moldova*. 2016. Vol. 24, N 3. P. 357–370.
6. Letichevsky A. Algebra of behavior transformations and its applications. In: Structural Theory of Automata, Semigroups, and Universal Algebra. NATO Science Series II. Mathematics, Physics and Chemistry. Kudryavtsev V.B., Rosenberg I.G. (Eds). Springer, 2005. Vol. 207. P. 241–272. https://doi.org/10.1007/1-4020-3817-8_10.
7. Letychevskyi O., Peschanenko V. Applying algebraic virtual machine to cybersecurity tasks. 2*022 IEEE 9th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)* (28–30 May 2022, Hammamet, Tunisia). Hammamet, 2022. P. 161–169. https://doi.org/10.1109/SETIT54465.2022.9875895.
8. Letichevsky A., Letychevskyi O., Peschanenko V., Weigert T. Insertion modeling and symbolic verification of large systems. In: SDL 2015: Model-Driven Engineering for Smart Cities. Fischer J., Scheidgen M., Schieferdecker I., Reed R. (Eds). *Lecture Notes in Computer Science*. Cham: Springer International Publishing, 2015. Vol. 9369. P. 3–18. https://doi.org/10.1007/978-3-319-24912-4_1.
9. Thomas R.W., DaSilva L.A., MacKenzie A.B. Cognitive networks. *Proc. First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005 (DySPAN 2005)* (08–11 November 2005, Baltimore, MD, USA, 2005). Baltimore, 2005. P. 352–360. https://doi.org/10.1109/DYSPAN.2005.1542652.
10. Hitzler P., Eberhart A., Ebrahimi M., Sarker M.K., Zhou L. Neuro-symbolic approaches in artificial intelligence. *National Science Review*. 2022. Vol. 9, Iss. 6. https://doi.org/10.1093/nsr/nwac035.

O. Letychevskyi

CONGNITIVE NETWORKS, THEIR PROPERTIES AND APPLICATIONS IN ATTACK DETECTION AND PREVENTION SYSTEMS

Abstract. Methods of real-time cyberattack detection, based on an algebraic approach, are considered. The method of algebraic matching is used, which is based on the methods of solving behavioral equations and algebraic modeling. At the same time, to increase the efficiency of cyberattack detection and prevention, it is suggested to use the composition of a neural network for the classification of attacks and the method of algebraic matching. This construction is called a cognitive system, whose concept was first formulated in 2005. The properties of the cognitive network, such as double classification and self-learning, are considered, and the technological line for detecting cyberattacks using cognitive networks is presented.

Keywords: artificial intelligence, neuron network, deep machine learning, behaviour algebra, neuro-symbolic approach, algebraic modeling, insertion modelling.

Надійшла до редакції 05.06.2023