КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 1 (21), 2023
ISSN 2663 - 4023
DOI 10.28925/2663-4023.2023.21.234251
УДК 004.89:[004.056:007]

Чичкарьов Євген
доктор технічних наук, професор, професор кафедри штучного інтелекту
Державний університет інформаційно-комунікаційних технологій, м. Київ, Україна
ORCID 0000-0002-4362-5129
chychkarovea@gmail.com

Зінченко Ольга
доктор технічних наук, доцент, завідувач кафедри штучного інтелекту
Державний університет інформаційно-комунікаційних технологій, м. Київ, Україна
ORCID 0000-0002-3973-7814
zinchenkoov@gmail.com

Бондарчук Андрій
доктор технічних наук, професор, директор навчально-наукового інституту інформаційних технологій
Державний університет інформаційно-комунікаційних технологій, м. Київ, Україна
ORCID 0000-0002-7309-4365
dekan.it@ukr.net

Асєєва Людмила
аспірант
Державний університет інформаційно-комунікаційних технологій, м. Київ, Україна
ORCID 0000-0001-5954-4211
aseewal@i.ua

МЕТОД ВИБОРУ ОЗНАК ДЛЯ СИСТЕМИ ВИЯВЛЕННЯ ВТОРГНЕНЬ З ВИКОРИСТАННЯМ АНСАМБЛЕВОГО ПІДХОДУ ТА НЕЧІТКОЇ ЛОГІКИ

Анотація. У дослідженні був запропонований новий метод побудови набору важливих ознак для вирішення задач класифікації. Цей метод заснований на ідеє використання ансамбля оцінювачів важливості ознак з підведенням підсумків і кінцевого результату ансамбля за допомо-гою алгоритмів нечіткої логіки. В якості оцінювачів важливості ознак було використано статистичні критерії (chi2, f_classif, коефіцієнт кореляції), критерій середнього зменшення помилок класифікації (mean decrease in impurity - MDI), критерій взаємної інформації (mutual_info_classif). Зменшення кількості ознак на усіх наборах даних впливає на точність оцінювання відповідно до критерію середнього зменшення помилок класифікації. Поки група ознак в на-борі даних для навчання містить перши за списком ознаки з найбільшим впливом, точність моделі знаходиться на початковому рівні, але при виключенні з моделі хоча б однієї з ознак з великим впливом, точність моделі помітно знижується. Найкращі результати класифікації для усіх досліджених наборів даних забезпечили класифікатори на основі дерев або найближчих сусідів: DecignTreeClassifier, ExtraTreeClassifier, KNeighborsClassifier. За рахунок виключення із моделі несуттєвих ознак досягається помітне збільшення швидкості навчання (до 60-70%). Для підвищення точності оцінювання було використано ансамблеве навчання. Найкращі показники за швидкістю навчання забезпечив класифікатор VotingClassifier, побудований на базі алгоритмів з максимальною швидкістю навчання. Для майбутньої роботи метою є подальше вдосконалення запропонованої моделі IDS в напрямках вдосконалення вибору класифікаторів для отримання оптимальних результатів, та налаштування параметрів вибраних класифікаторів, удосконалення стратегії узагальнення результатів окремих класифікаторів. Для запропонованої моделі істотний інтерес представляє можливість виявлення окремих типів атак з урахуванням багатокласового прогнозування.

Ключові слова: система виявлення вторгнень, машинне навчання, ансамблеве навчання, класифікатор, нечітка логіка, кібератака; кіберзахист з використанням машинного навчання; алгоритми обрання ознак

ВСТУП

Виявлення вторгнень є важливою частиною мережевої безпеки боротьби з незаконним доступом до мережі чи зловмисними кібератаками. Щоб протидіяти цім атакам, було розроблено багато інструментів і механізмів для мережевої та хмарної безпеки, включаючи різні системи виявлення вторгнень (intrusion detection system - IDS).

Впровадження методів машинного навчання у розробку IDS широко вивчалося протягом останнього десятиріччя. Моделі машинного навчання (ML) показали перспективні результати в прогнозуванні або класифікації даних у багатьох областях досліджень, і в контексті IDS ML використовується для класифікації того, чи є трафік безпечним чи атакуючим [1].

Залежно від джерела інформації, IDS можна розділити на IDS на базі хоста (HIDS) та IDS на основі мережі (NIDS). HIDS пов'язані з інформацією операційної системи (системними викликами та ідентифікаторами процесів), тоді як NIDS виконують аналіз мережевих подій (IP-адреси, протоколи, сервісні порти, обсяг трафіку тощо) [2]. На основі виконуваного аналізу IDS можна розділити на IDS на основі сигнатур (на основі неправомірного використання) та IDS на основі аномалій. У signature-based IDS (SIDS) підтримується база даних відомих сигнатур атак, і IDS зіставляє проаналізовані дані з записами бази даних, щоб виявити вторгнення. Такі системи найкраще підходять для виявлення відомих атак, але не здатні виявляти нові типи атак (раніше невідомі) [3]. Навпаки, IDS на основі аномалій (AIDS) намагається зрозуміти нормальну поведінку системи та встановлює граничне значення. Коли це спостереження в якийсь момент відхиляється від нормальної поведінки, перевищуючи задане граничне значення, видається сигнал про аномалію. Оскільки IDS на основі аномалій намагається виявити підозрілі події, це гарне рішення для виявлення раніше невидимих атак. Однак рівень хибно-позитивних результатів виявлення вторгнень вище при AIDS, ніж при SIDS [2, 4].

Технологія машинного навчання дозволяє IDS навчатися та покращувати продуктивність системи шляхом аналізу попередніх даних. Методика виявлення вторгнень є основою багатьох типів дослідницьких робіт, оскільки рівень виявлення та точність моделей машинного навчання не відповідають вимогам для класифікації вторгнень. Більше того, багато рішень є недостатньо ефективними для великої кількості даних з використанням повного набору даних [5]. Крім того, безліч робіт з виявлення вторгнень було проведено з використанням набору даних NSL-KDD або KDD99, який зараз вважається застарілим для виявлення сучасних кібератак [6].

Набори даних, які використовують для навчання систем виявлення вторгнень, відносно великі і мають досить велику розмірність простору ознак. Більшість досліджень використовують кілька моделей машинного навчання та кілька наборів даних для оцінки.

Для великого набору даних важливо обрати метод вибору ознак для виключення непотрібних ознак та використовувати лише набір важливих ознак як на етапі навчання, так і на етапі тестування. Вибір ознак - це процес зменшення кількості вхідних змінних при побудові прогнозної моделі. Оскільки IDS має справу з даними, які мають нерелевантні або надмірні ознаки, зменшення кількості ознак не лише зменшує обчислювальні витрати, а й покращує показник точності виявлення. Однак різні підходи до вибору набору релевантних ознак дають підмножини ознак, які не збігаються.

Наприклад, в роботі [7] для побудови моделі було обрано набір ознак, які найчастіше повторювалися в усіх методах обрання ознак.

Постановка проблеми.

Таким чином, вивчення методів вибору ознак у даних мережевого трафіку з метою виявлення потенційних атак є актуальним завданням.

Відомі набори даних для навчання систем виявлення вторгнень, містять ознаки, які можуть бути необов'язковими або нерелевантними.

Метою цієї роботи є оцінка різних існуючих методів вибору атрибутів і розробка нового методу з точки зору швидкості виявлення, точності і обчислювальної продуктивності з урахуванням двійкової та багатокласової класифікації можливих атак.

У запропонованому нижче підході була розроблена модель вибору релевантного набору ознак на основі оцінки важливості за декількома критеріями (Gini impurities або статистичних оцінок) з використанням нечітких правил для відбору важливих та релевантних ознак.

Аналіз останніх досліджень і публікацій.

За останні роки було запропоновано багато різних методів виявлення вторгнень, в тому числі з використанням машинного або глибокого навчання [8].

В роботі [9] представлено підхід для IDS на основі машинного навчання, в якому поєднано класифікатор дерева рішень (DT). Автори проаналізували набір даних NSL-KDD. За допомогою методу відбору ознак на основі фільтрів було обрано 14 значущих ознак, це дослідження проводилося як для задач бінарної, так і багатокласової класифікації.

В роботі [10] для виявлення атак типу "розподілена відмова в обслуговуванні" (DDoS) було використано кілька фільтрів: хі-квадрат, приріст інформації, коефіцієнт посилення та алгоритм ReliefF для вибору оптимальної кількості ознак. Для аналізу продуктивності системи вони навчили та оцінили моделі на наборі даних NSL-KDD, було обрано 13 важливих ознак.

Автори [11] провели моделювання на наборах даних UNSW-NB15, NSL-KDD та KDDCup99. За допомогою роєвого інтелекту було обрано 10 об'єктів з KDDCup99, 14 об'єктів з набору даних UNSW-NB та 18 об'єктів з NSL-KDD, але для обрання ознак було використано алгоритми роєвого інтелекту.

В роботі [12] реалізували п'ять контрольованих моделей з використанням методу Extreme Gradient Boosting (XGBoost) для зменшення кількості ознак з 42 до 19 (з урахуванням їх важливості). При зменшенні кількості ознак для набору даних UNSW-NB15 автори [12] спостерігали збільшення точності моделі DecisionTree з 88,13 до 90,85% при виконанні завдання двійкової класифікації.

Багато алгоритмів вибору ознак у випадку багатокласової класифікації ґрунтуються на взаємній інформації [13].

Побудова релевантного набору ознак на основі оцінки їх важливості використовується для навчання моделей машинного навчання. Аналіз важливості ознак грає істотну роль і для пояснення результатів машинного навчання [14, 15].

Наприклад, в роботі [16] було запропоновано двоетапний підхід до виявлення вторгнень. На першому етапі виконувалася бінарна класифікація. На другому етапі вихідні дані про атакуючий трафік передавалися в мультикласові класифікатори для ідентифікації кожної атаки. Для завдання бінарної класифікації автори [16] використовували метод вибору ознак на основі оцінки важливості після побудови моделі

RandomForest.

Але різні моделі машинного навчання можуть генерувати різні оцінки важливості ознак через відмінності у їх алгоритмах навчання [15]. На думку [15], лінійні моделі генерують чіткі оцінки важливості ознак за допомогою лінійних співвідношень. Для суттєво нелінійних моделей оцінки важливості ознак можуть бути локальними, що залежать від властивостей поверхні відгуку.

Покращення надійності процедури вибору ознак можливе за рахунок використання ансамблевого методу оцінки важливості ознак (ensemble feature importance – EFI) [15]. У роботі [17] для оцінки важливості ознак запропоновано підхід, який базується на введенні характеристик (низька, середня та висока важливість) для побудови нечіткої функції належності ознак до класу важливих. На думку авторів [17], нечіткі системи мають додаткову перевагу, пояснюючи оцінки важливості ознак лінгвістичними термінами, що полегшує для нефахівців розуміння моделей. За результатами розрахунків [17] з використанням різних синтетичних наборів даних із різним рівнем кількості ознак, нечітка ансамблева оцінка важливості ознак забезпечила більш точні результати порівняно з точними середніми або ансамблевими підходами на основі більшості голосів.

Зростання застосування машинного навчання у дедалі більшій кількості важливих додатків призвело до появи складних та ефективних рішень практично без втручання людини. Для систем, критично важливих для безпеки, розуміння того, як генеруються вихідні дані машинного навчання, особливо важливим є для перевірки та діагностики моделі [17-18], розробки наступного покоління інтелектуальних систем.

Таким чином, використання ансамблевого підходу і методів інтерполяції нечітких правил забезпечує більш точні і надійні результати обрання ознак і загальні можливості системи виявлення вторгнень.

Мета дослідження

Мета дослідження – дослідження використання ансамблевого підходу для оцінки важливості ознак, уточнення можливості використання системи нечітких правил для узагальнення результатів ансамблю, побудова моделі вибору набору ознак для навчання системи виявлення вторгнень.

ТЕОРЕТИЧНІ ОСНОВИ ДОСЛІДЖЕННЯ

Методи вибору ознак

У цьому розділі ми представляємо основні концепції вибору ознак. Залежно від взаємодії з моделлю класифікації методи вибору ознак можна розділити на методи фільтрації, обгортки та вбудовані методи [19].

Відомі також гібридні та ансамблеві методи вибору ознак.

Гібридний підхід поєднує два різних методи для використання переваг обох підходів, при цьому загальною комбінацією є методи фільтра і оболонки.

Техніка ансамблю об'єднує ансамбль методів обрання ознак або підмножин даних та ознак, потім тім чи іншим способом формує загальний результат.

Метод фільтру

Методи типу фільтра вибирають функції, оцінюючи внутрішні властивості даних з урахуванням статистичних показників, а чи не ефективності перехресної перевірки.

Їх легко масштабувати стосовно багатовимірних наборів даних незалежно від алгоритму навчання; вони відносно прості та швидкі у обчисленнях; і вони стійкі до перенавчання. У цьому методі кожному об'єкту надається бал, який визначається обраним статистичним методом. Після цього всі функції ранжуються в порядку зменшення, а об'єкти з низькими оцінками видаляються з використанням порогового значення. Інші функції становлять підмножина функцій і потім вводяться у модель класифікації. Отже, вибір ознак проводиться один раз, а потім можна використовувати різні класифікатори. Цей підхід має два основних недоліки: ознаки вибираються незалежно від класифікатора та ігнорування залежності ознак.

Деякі поширені статистичні показники, що використовуються в цьому методі, є приріст інформації (IG), кореляція Пірсона (PS), хі-квадрат ($\chi 2$), взаємна інформація (MI) і симетрична невизначеність (SU).

Метод оболонки

У цій методології визначається стратегія пошуку можливих підмножин ознак і алгоритм навчання навчається з використанням цих підмножин ітеративним способом. На відміну від методів фільтрації, методи-оболонки взаємодіють із класифікатором, проте оцінка підмножин ознак виходить із використання конкретної моделі класифікації, що робить цей метод специфічним для моделі навчання. Цей метод надає неоптимальні підмножини ознак для навчання моделі, оскільки оцінка всіх можливих підмножин недоцільна з обчислювальної точки зору і зазвичай дає кращу точність прогнозування ніж методи фільтрації, але вимагає великих обчислювальних ресурсів через накладні витрати на пошук і залежність учня.

Пошук для створеня підмножин може здійснюватись за допомогою таких схем, як прямий вибір, зворотний вибір, покроковий вибір або евристичний пошук. Спочатку додається ознака з найбільшим вкладом (найкращою оцінкою). Потім вибирається інша, важливіша функція, що забезпечує найкращу продуктивність (performance) разом із раніше доданою функцією. Цей процес триває доти, доки включення нової ознаки не перестане покращувати продуктивність класифікатора. При зворотному виключенні алгоритм починається з усіх доступних ознак і відкидає рекурсивно найбільш незначні ознаки з моделі. Цей процес виключення повторюється доти, доки видалення ознак не перестане покращувати продуктивність моделі. Для покрокового вибору цей метод є комбінацією прямого вибору та зворотного виключення. Він починається з порожнього набору, і кожної ітерації додається найважливіша функція. При додаванні нової ознаки раніше вибрані ознаки видаляються, якщо якась із них стала незначною. Евристичний пошук пов'язаний з оптимізацією та спрямований на оптимізацію цільової функції в оцінці різних підмножин [20].

Вбудований метод

Цей метод включає переваги методів фільтра і оболонки і одночасно виконує побудову набору ознак і моделі. Як і методи-оболонки, вони специфічні для моделі навчання, але мають меншу обчислювальну складність, ніж методи-оболонки [19].

Ще один метод інтеграції алгоритму відбору ознак створення моделі — це дерева рішень. Ці деревоподібні методи є непараметричними моделями, в яких об'єкти розглядаються як вузли. Деревоподібні стратегії, що використовуються випадковими лісами, накопичують різну кількість дерев рішень і ранжують вузли (тобто ознаки) щодо зменшення impurity (наприклад Gini impurity) по всіх деревах, наприклад, дереву класифікації та регресії (CART) [21].

Традиційні підходи вибору ознак мають низку недоліків. Наприклад, методи фільтрації оцінюють значущість кожної ознаки індивідуально, не зважаючи на відносини та взаємодії між ознаками. Методи-обгортки можуть забезпечити оптимальне підмножина ознак, але їх складність робить їх недосконалими. Вони не є кращими, особливо у комбінаторних методах, таких як ансамблеві методи. Крім того, вони не застосовуються до даних з невеликою кількістю вибірок через перенавчання. Вбудовані методи, такі як оболонки, специфічні для моделі, тому можуть надавати інший підбір ознак для одного і того ж набору даних. Основним недоліком таких методів є їх нездатність ефективно видаляти надлишкові ознаки та ефективно зберігати інформативні ознаки [22-23].

Використані набори даних

CSE-CIC-IDS2018 - це загальнодоступний набір даних про вторгнення [24]. Цей набір даних було створено з урахуванням недоліків попередніх наборів даних про вторгнення. CSE-CIC-IDS2018 — це один із найбільших наборів даних IDS із реальним мережевим трафіком і широким спектром атак. Він також містить звичайні дані та дані про вторгнення. CICIDS2018 включає сім різних сценаріїв атак: Brute-force (Web, XSS, FTP, SSH), SQL Injection і DDoS (HOIC, LOIC- UDP, LOIC-HTTP), Heartbleed, Botnet, DoS (Hulk, SlowHTTPTest, GoldenEye, Slowloris), DDoS (HOIC, LOIC- UDP, LOIC-HTTP), Web attacks, і проникнення в мережу зсередини.

Набір даних KDDCup99 [25] містить 41 ознаку та охоплює чотири основні категорії атак: атаки зондування (атаки зі збором інформації), атаки на відмову в обслуговуванні (DoS), атаки користувача на root (U2R), атаки віддаленого до локального (R2L), спостереження та інші зондування, наприклад, сканування портів (probing).

NSL-KDD — це поновлена версія набору даних KDDCup99 [26]. Це ефективний контрольний набір даних, який допоможе дослідникам порівняти різні методи виявлення вторгнень. Цей набір даних не має надлишкових записів, тому будь-яка модель, навчена на цьому наборі даних, не повинна бути схильна до повторних записів атак. Загалом у цьому наборі даних для одного запису є 43 ознаки. З 43 ознак 41 пов’язана з вхідним трафіком, а дві інші є мітками та балами вхідного трафіку. Цей набір даних має загалом чотири класи для різних атак: зондування, атак користувача на root (U2R), відмова в обслуговуванні (DoS) і віддалена локальна атака (R2L).

Набір даних UNSW-NB15 містить понад два мільйони записів, 48 ознак і дев’ять різних типів атак: Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode та Worm [27].

Набір даних LITNET-2020 — це відносно новий набір даних, зібраний академічною мережею LITNET (Литовська науково-освітня мережа) у мережевому трафіку Литви в режимі реального часу. Це реальний і сучасний мережевий набір даних на основі потоку [28], розроблений для тестування систем IDS. У цьому наборі даних було 85 функцій мережевого потоку та 12 типів атак.

МЕТОДИКА ДОСЛІДЖЕННЯ

Усі розрахунки в роботі було виконано на комп'ютері з процесором Intel Core i5 з 8 ГБ пам’яті під управлінням Windows 11. Було використано Python 3.11.6 та пакет Scikit-Learn 1.3.1.

Методи відбору ознак

Сучасні IDS повинні відповідати вимогам і зростаючим потребам у вдосконаленні

технологій. Для успішної роботи IDS необхідна високоефективна класифікація з використанням даних, які раніше не були відомі системі. IDS зазвичай обробляють досить великий обсяг даних, які містять різні надлишкові ознаки, що призводить до низького рівня точності та тривалого часу обробки [29]. Це робить вибір ознак для класифікації важливим питанням. Для скорочення часу навчання моделі класифікації та підвищення рівня її точності важливим питанням є вибір найважливіших ознак із набору даних. У цьому дослідженні досліджувалися різні методи вибору ознак і ансамблю, щоб створити ефективну IDS з високою точністю розпізнавання загроз.

Попередня обробка даних передбачає наступні операції:

– виявлення/усунення невідповідностей,
– виправлення помилок в даних,
– масштабування та нормалізація.

Після попередньої обробки були обрані найважливіші ознаки в наборі даних за допомогою відповідних алгоритмів. Час начання та класифікації та інші показники класифікатора оцінювались разом із точністю результатів.

Дані про вторгнення з відомих наборів даних та реальні дані трафіку в корпоративних мережах мають деякі особливості:

– в одній виборці зазвичай присутні ознаки різних типів – числові та категоріальні;
– для більшості аналізованих наборів даних розглядається як бінарна, так і багатокласова класифікація;
– кількість ознак варіюється від вибірки до вибірки;
– зв'язок відгуку та окремих ознак може бути як лінійним, так і нелінійним, або практично відсутнім.

Для вибору ознак було використані декілька варіантів обрання ознак:

– чисто статистичні методи оцінки важливості ознак, розраховані на линійний зв’язок ознак та класу відгуку (тест Chi2; F-classif; Correlation);
– метод з використанням оцінки важливості ознак за критерієм середнього зменшення помилок класифікації (mean decrease in impurity – MDI; для розрахунку важливості використовували RandomForestClassifier або ExnraTreeClassifier);
– метод оцінки важливості ознак на взаємної інформації (mutual information).

В якості окремих класифікаторів ансамбля використовувались декілька вариантів (вони забезпечували найшвидше навчання моделі):

– алгоритм ExtraTreeClassifier;
– алгоритм DesignTreeClassifier;
– алгоритм k найближчих сусідів (K nearest neibour).

Для узагальнення результатів ансамблевого класифікатора використовувався алгоритм VotingClassifier (як у варианті звичайного голосування, так і в варианті з використанням сукупності нечітких правил). В якості мета-класифікатора було використано також алгоритм RandomForestClassifier.

Пропонований метод нечіткого обрання важливих ознак

Припустимо, що набір даних, для якого необхідно побудувати оцінку важливості ознак, містить M ознак. Усі ознаки перетворюються на числові та нормалізуються.

Для оцінки важливості ознак використовуємо N методів, кожен з яких надає оцінку важливості ознаки j ($j \in 1 \dots M$) з використанням метода i ($i \in 1 \dots N$), яка дорівнює f_j^i.

Введемо універсальну множину U з переліком граничних значень рівня важливості ознаки від 0 до 1 (наприклад, важливість ознаки j при оцінці методом i висока, якщо $f_j^i >$

0.8). Універсальна множина буде містити 3 (низька, помірна і висока важливість, як в роботах [8,9]) або 5 рівнів (важливість висока, достатньо висока, середня, достатньо низька, низька).

Для результуючої оцінки важливості ознак за усіма методами було використовується алгоритм Мамдані [30]. У кінцевий набір даних включаються ознаки, важливість яких на рівні висока та досить висока.

Для підвищення надійності і узагальнення методу оцінка важливості ознак може виконуватись для вибірок із початкового набору даних із подальшим обчисленням середнього результату або за допомогою алгоритму голосування.

Підготовка даних

Деяки набори даних, яки було використано, містили кілька окремих файлів з даними. Вони об'єднувались в один результуючий документ. Рядки з помилковими даними видалялись. Стовпчики з категоріальними ознаками перетворювались на цифрові. Значення true, false, off і low були в результаті перетворені на нуль або одиницю.

Більшість алгоритмів класифікації функціонують більш ефективно, коли ознаки мають порівнянну величину, оскільки це допомагає зменшити зміщення в бік ознак із високими значеннями множинності в результатах прогнозування. Отримані числові дані нормалізувалися за допомогою вбудованих можливостей scikit-learn (переважно StandardScaler).

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Основні результати роботи одиничних класифікаторів наведені в таблиці 1. Позначки в таблиці: KNN – класифікатор KNeibourClassifier (метод найближчих сусідів); DTC – класифікатор DecignTreeClassifier; ETC – класифікатор ExtraTreeClassifier; RFC-класифікатор RandomForestClassifier; MLP – класифікатор MLPClassifier (багатошаровий персептрон); ADA - класифікатор ADABoost; LR – логістична регресія.

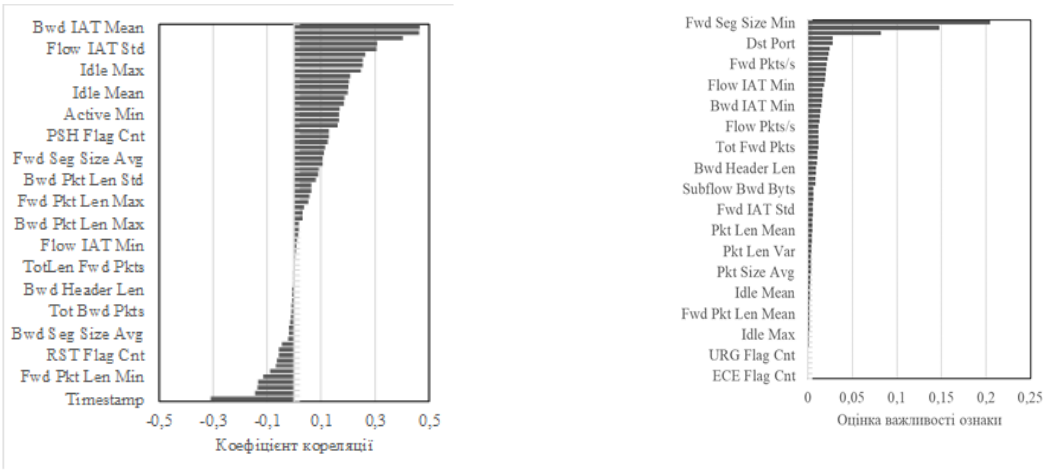
Таблиця 1

Результати без недостатньої вибірки та вибору ознак (IDS2018)

Класифікатор	Точність, %	Показник F1	Час навчання, с
KNN	99.99	0,999	0.0945
DTC	100,00	1.00	1,836
ETC	100,00	1.00	0,214
RFC	100,00	1.00	22,01
MLP	99.5	0,995	206,67
ADA	68,0	0,716	37,48
LR	94,9	0,952	54,31

Результати показують, що алгоритми, засновані на дереві, є досить успішними при проведенні класифікації різних вибірок, які було використано для побудови класифікатора IDS. Досить точний швидкий результат отримано також з використанням алгоритму k найближчих сусідів.

Критерії для обрання набору ознак надають досить неоднозначні результати (рис. 1). Відомі методи обрання ознак, які використовуються по одному, нерідко дають різні результуючі набори даних.

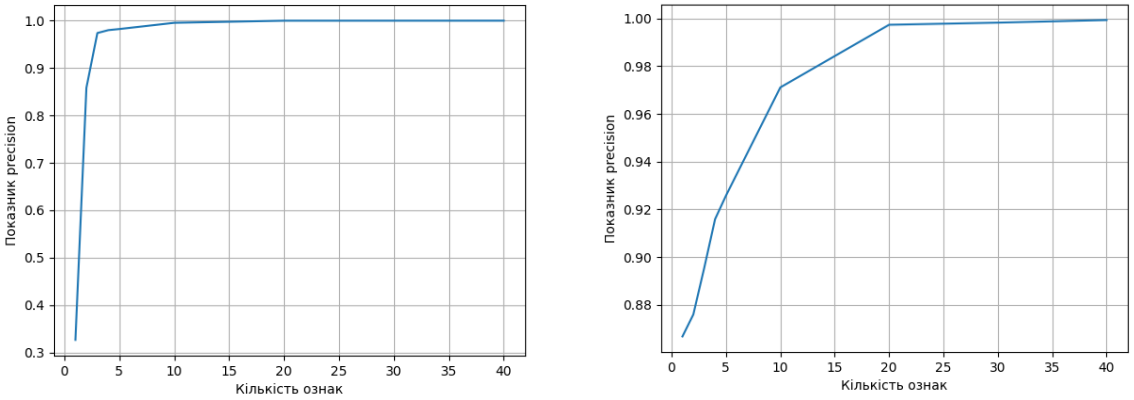


а) Кореляційна діаграма наявності атаки і ознак набору даних CICIDS2018

б) Діаграма MDI для ознак набору даних CICIDS2018

Рис. 2. Варианти критеріїв обрання ознак для набору даних CICIDS2018

Приклад обчислення корреляції ознак набору даних CICIDS2018 і наявності атаки наведено на рис. 1а. Після аналізу поведінки компонентів пропонованої системи було додано ще один вариант обрання ознак, більш адекватний саме для класифікаторів на основі дерев: вибір ознак за критерієм середнього зменшення помилок класифікації (mean decrease in impurity - MDI). Приклад обчислення вибору ознак за критерієм середнього зменшення помилок класифікації для набору даних CICIDS2018 наведено на рис. 1б. Як видно з наведених рисунків, кількість ознак у результуючому наборі даних для навчання класифікатора залежить від критерію вибору цих ознак. Те ж саме питання виникає й для обрання ознак за допомогою тестів chi-square або fuzzy logic. При цьому точність класифікації за таким зменшеним набором не змінюється, але пояснення, чому саме такий набор даних дає можливість відслідкувати вторгнення, метод обрання ознак не надає. Приклад змінення точності класифікації даних KDD99 та NSL-KDD в залежності від кількості ознак, які включено в модель, наведено на рис. 2 (для побудови цих рисунків відбір ознак було здійснено за критерієм f_classif).



а) набор даних KDD99

б) набор даних NSL-KDD

Рис. 2. Зростання точності класифікації моделі в залежності від частки кількості використаних ознак

Кількість ознак, які забезпечують досить високу точність класифікації, складає 25-50% вихідного набору даних. Приклад результатів тестування моделі зі скороченою кількістю ознак наведено в таблиці 2.

Таблиця 2

Результати для набору даних зі скороченой кількістю ознак

Класифікатор	Точність,%	Показник F1	Час навчання, с
KNN	99.99	0,999	0,0628
DTC	100,00	1.00	1,225
ETC	100,00	1.00	0,177
RFC	100,00	1.00	28,38
MLP	99.6	0,995	202,15
ABC	94,2	0,940	44,81
LR	93,4	0,936	61,95

Алгоритми LogisticRegression, AdaBoost, MultiLayerPerceptron довго навчаються і для досягнення високих показників точності потрібують налагодження. Коли час прогнозування та рівень точності оцінюються разом, було знайдено, що алгоритми дерева рішень (DTC), найближчих сусідів (KNN) та додаткових дерев (ETC) дають результати з високою точністю та швидким часом прогнозування.

Показники точності та час роботи цих алгоритмів у результаті роботи з оригінальними наборами даних і наборами даних зі скороченою кількістю ознак наведені в таблиці 3. У цій таблиці представлені дані KDDCup99, хоча аналогічні результати були отримані й для інших наборів даних – LITNET, ISD2018.

Таблиця 3

Порівняння точності і часу навчання моделі на наборах даних з різною кількістю ознак (вихідний набор даних KDDCup99)

Параметр	Класифікатор		
	KNN	DTC	ETC
Точність, %			
41 ознака (вихідний)	99,9	100	100
21 ознака	99,9	100	100
Час навчання, с			
41 ознака (вихідний)	0,0945	1,836	0,214
21 ознака	0,0628	1,225	0,177

Якщо слідкувати за переліком ознак, які були обрані за важливістю, то перелік обраних ознак залежить від метода оцінки. Приклад послідовності включення ознак в результуючий набір з використанням двох критеріїв наведено в таблиці 4.

Таблиця 4

Результати обрання ознак за важливістю з використанням різних критеріїв (набір даних KDD99)

Кількість ознак	Критерій f_classif	Критерій MDI	Критерій mutual_info_classif
1	['wrong_fragment']	['count']	['count']
2	['wrong_fragment' 'srv_count']	['count' 'same_srv_rate']	['src_bytes' 'count']
3	['wrong_fragment' 'srv_count'	['service' 'serror_rate'	['service' 'src_bytes'

	'same_srv_rate']	'dst_host_same_srv_rate']	'count']
4	['wrong_fragment' 'count' 'srv_count' 'same_srv_rate']	['srv_serror_rate' 'dst_host_count' 'dst_host_srv_count' 'dst_host_same_src_port_rate']	['service' 'src_bytes' 'count' 'dst_host_same_src_port_rate']
5	['protocol_type' 'wrong_fragment' 'count' 'srv_count' 'same_srv_rate']	['protocol_type' 'srv_serror_rate' 'diff_srv_rate' 'dst_host_srv_count' 'dst_host_srv_rerror_rate']	['service' 'src_bytes' 'count' 'srv_count' 'dst_host_same_src_port_rate']

Для побудови таблиці 4 було використано три критерії важливості ознак – критерій F-значення ANOVA (f_classif), критерій середнього зменшення помилок класифікації (mean decrease in impurity - MDI), критерій взаємної інформації (mutual_info_classif). Критерії взаємної інформації або важливості перестановки (permutation importance) розраховуються досить повільно, тому для обрання ознак за запропонованим підходом не використовувались.

Деяке підвищення точності оцінювання було досягнуто за рахунок використання метакласифікаторів (VotingClassifier, StackingClassifier, RandomForestClassifier). Було побудовано декілька моделей з використанням різних алгоритмів класифікації, які навчались на підмножинах зі зменшеною кількістю ознак, або на наборах даних вихідного розміру.

Вибір результатів класифікації встановлювався переважно за допомогою алгоритму VotingClassifier. Його ідея полягає в тому, щоб поєднати концептуально різні класифікатори машинного навчання та використовувати більшість голосів або середні передбачені ймовірності (м'яке голосування) для прогнозування міток класу. Такий класифікатор може бути корисним для набору однаково ефективних моделей, щоб збалансувати їх окремі слабкі сторони.

Результати навчання і використання моделі ансамблю наведено в таблиці 5. Для побудови таблиці було використано або повні набори даних, або скорочені набори даних, які було побудовано з використанням запропонованого підходу. Як випливає з таблиці 5, точність класифікації при навчанні моделі на скороченому наборі даних з обраними важливими ознаками практично не знижується.

Таблиця 4

Результати оцінювання точності різних ансамблевих класифікаторів і наборів даних

Набір даних	Кількість ознак	Класифікатор	Точність, %	Час навчання, с
KDDCup99	41	RF	99,55	22,55
	20	RF	100,0	12,57
	20	Voting	99,34	2.26
NSL-KDD	41	RF	99,99	9,53
	16	RF	100,0	9,18
	16	Voting	99,34	1,31
UNSW-NB15 (скорочений набір)	44	RF	99,99	17,66
	21	RF	99,99	12,40
	21	Voting	99,34	2,01
IDS 2018 (одна доба)	79	RF	99,99	172.1
	21	RF	99,99	121,4
	21	Voting	99,99	8,51

Час навчання Voting-класифікатора залежить від базових класифікаторів, яки використано для його побудови. Наприклад, якщо для побудови вотуючого класифікатора користуватись RandomForestClassifier (та ще два інші) і порівнювати швидкість навчання із безпосередньо RandomForestClassifier, то час навчання вотуючого класифікатора зростає на 10-15 %. Якщо порівнювати, наприклад, швидкість навчання вотуючого класифікатора на базі KNearestNeiborClassifier, ExtraTreeClassifier, DesignTreeClassifier з навчанням RandomForestClassifier, то час навчання вотуючого класифікатора значно менше (в залежності від набору даних, який обрано для навчання, але щонайменше на 60-70%).

Вплив кількості ознак, на час навчання класифікаторів і ансамбля VotingClassifier в цілому наведено на рис. 3. Для ExtraTreeClassifier час навчання практично не залежить від кількості ознак. Для DesignTree або KNeibors (і, як наслідок, для класифікатора Voting в цілому) час навчання помітно зростає зі збільшенням кількості ознак.

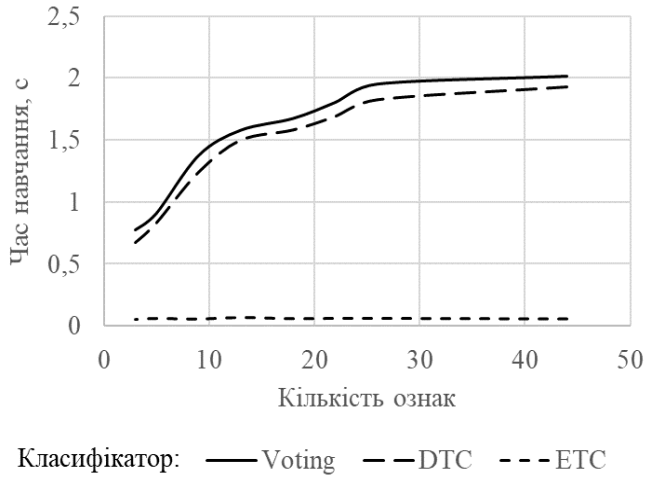


Рис. 3. Залежність часу навчання деяких класифікаторів ансамблю від кількості ознак (навчання на частині набору даних UNSW-NB15)

Для зменшення дисперсії базового оцінювача та підвищення надійності класифікації було використано алгоритм bagging, який агрегує індивідуальні прогнози за випадковими підмножинами початкового навчального набору для формування остаточного прогнозу.

При відповідному налаштуванні точність класифікацій трохи збільшується, але обрання Stacking або Bagging класифікатора як основи навчання моделі для усіх наборів даних збільшує час навчання більш ніж на порядок (в залежності від базових класифікаторів або кількості підмножин даних). При збільшенні кількості спостережень в наборі даних для навчання ефект зростання часу навчання стає більш помітним.

Спроба обрання ознак з використанням bagging-підходу призвела до аналогічного результату: час побудови набору ознак помітно збільшується.

Таким чином, запропоновано новий спосіб побудови множини ознак з використанням ансамблю методів визначення їх важливості, де кінцева оцінка підводиться за допомогою алгоритмів нечіткої логіки. Отримані результати дозволила підтримати високий рівень точності (краще 99%) і низький рівень помилок після навчання моделей IDS на наборах даних зі зменшеною кількістю ознак.

ВИСНОВКИ

У дослідженні був запропонований новий метод побудови набору важливих ознак для вирішення задач класифікації. Цей метод заснований на ідеє використання ансамбля оцінювачів важливості ознак з підведенням підсумків і кінцевого результату ансамбля за допомогою алгоритмів нечіткої логіки. В якості оцінювачів важливості ознак було використано статистичні критерії (chi2, f_classif, коефіцієнт кореляції), критерій середнього зменшення помилок класифікації (mean decrease in impurity - MDI), критерій взаємної інформації (mutual_info_classif).

Зменшення кількості ознак на усіх наборах даних впливає на точність оцінювання відповідно до критерію середнього зменшення помилок класифікації. Поки група ознак в наборі даних для навчання містить перши за списком ознаки з найбільшим впливом, точність моделі знаходиться на початковому рівні, але при виключенні з моделі хоча б однієї з ознак з великим впливом, точність моделі помітно знижується.

Найкращі результати класифікації для усіх досліджених наборів даних забезпечили класифікатори на основі дерев або найближчих сусідів: DecignTreeClassifier, ExtraTreeClassifier, KNeighborsClassifier.

За рахунок виключення із моделі несуттєвих ознак досягається помітне збільшення швидкості навчання (до 60-70%). Для підвищення точності оцінювання було використано ансамблеве навчання. Найкращі показники за швидкістю навчання забезпечив класифікатор VotingClassifier, побудований на базі алгоритмів з максимальною швидкістю навчання.

Для майбутньої роботи метою є подальше вдосконалення запропонованої моделі IDS в напрямках вдосконалення вибору класифікаторів для отримання оптимальних результатів, та налаштування параметрів вибраних класифікаторів, удосконалення стратегії узагальнення результатів окремих класифікаторів. Для запропонованої моделі істотний інтерес представляє можливість виявлення окремих типів атак з урахуванням багатокласового прогнозування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Chua, T.-H., & Salam, I. (2023). Evaluation of Machine Learning Algorithms in Network-Based Intrusion Detection Using Progressive Dataset. Symmetry, 15(6), 1251. https://doi.org/10.3390/sym15061251
2. Disha, R. A., & Waheed, S. (2022). Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique. Cybersecurity, 5(1). https://doi.org/10.1186/s42400-021-00103-8
3. Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity, 2(1). https://doi.org/10.1186/s42400-019-0038-7
4. Liao, H.-J., Richard Lin, C.-H., Lin, Y.-C., & Tung, K.-Y. (2013). Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 36(1), 16–24. https://doi.org/10.1016/j.jnca.2012.09.004
5. Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks. IEEE Access, 5, 21954–21961. https://doi.org/10.1109/access.2017.2762418
6. Divekar, A., Parekh, M., Savla, V., Mishra, R., & Shirole, M. (2018). Benchmarking datasets for Anomaly-based Network Intrusion Detection: KDD CUP 99 alternatives. У 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS). IEEE. https://doi.org/10.1109/cccs.2018.8586840
7. Alkasassbeh, M. (2017). An empirical evaluation for the intrusion detection features based on machine learning and feature selection methods. https://doi.org/10.48550/arXiv.1712.09623

8. Catania, C. A., & Garino, C. G. (2012). Automatic network intrusion detection: Current techniques and open issues. Computers & Electrical Engineering, 38(5), 1062–1072. https://doi.org/10.1016/j.compeleceng.2012.05.013
9. Ingre, B., & Yadav, A. (2015). Performance analysis of NSL-KDD dataset using ANN. In: 2015 International Conference on Signal Processing And Communication Engineering Systems (SPACES). IEEE. pp 92–96. https://doi.org/10.1109/spaces.2015.7058223
10. Osanaiye, O., Cai, H., Choo, K.-K. R., Dehghantanha, A., Xu, Z., & Dlodlo, M. (2016). Ensemble-based multi-filter feature selection method for DDoS detection in cloud computing. EURASIP Journal on Wireless Communications and Networking, 2016(1):1-10. https://doi.org/10.1186/s13638-016-0623-3
11. Liu, H., Yan, X., & Wu, Q. (2019). An Improved Pigeon-Inspired Optimisation Algorithm and Its Application in Parameter Inversion. Symmetry, 11(10), 1291. https://doi.org/10.3390/sym11101291
12. Kasongo, S. M., & Sun, Y. (2020). Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset. Journal of Big Data, 7(1). https://doi.org/10.1186/s40537-020-00379-6
13. Wang, X., & Zhou, Y. (2022). Multi-Label Feature Selection with Conditional Mutual Information. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.4153295
14. Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. Evaluating feature importance estimates, 2018. https://doi.org/10.48550/arXiv.1806.10758
15. Rengasamy, D., Rothwell, B. C., & Figueredo, G. P. (2021). Towards a More Reliable Interpretation of Machine Learning Outputs for Safety-Critical Systems Using Feature Importance Fusion. Applied Sciences, 11(24), 11854. https://doi.org/10.3390/app112411854.
16. Souhail et. al., M. (2019). Network Based Intrusion Detection Using the UNSW-NB15 Dataset. International Journal of Computing and Digital Systems, 8(5), 477–487. https://doi.org/10.12785/ijcds/080505
17. Rengasamy, Divish & Mafeni Mase, Jimiama & Rothwell, Benjmain & Torres, Mercedes & Alexander, Morgan & Winkler, David & Figueredo, Grazziela. (2022). Feature Importance in Machine Learning Models: A Fuzzy Information Fusion Approach. Neurocomputing. 511. https://doi.org/10.1016/j.neucom.2022.09.053.
18. Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion, 58, 82–115. https://doi.org/10.1016/j.inffus.2019.12.012
19. Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, and Huan Liu. (2017). Feature Selection: A Data Perspective. ACM Comput. Surv. 50, 6, Article 94 (November 2018), 45 pages. https://doi.org/10.1145/3136625
20. Huan Liu and Lei Yu. (2005). Toward Integrating Feature Selection Algorithms for Classification and Clustering. IEEE Trans. on Knowl. and Data Eng. 17, 4 (April 2005), 491–502. https://doi.org/10.1109/TKDE.2005.66
21. Breiman, L. (2017). Classification and Regression Trees (1st ed.). Routledge. https://doi.org/10.1201/9781315139470
22. Khaire, Utkarsh & Dhanalakshmi, R.. (2019). Stability of Feature Selection Algorithm: A Review. Journal of King Saud University - Computer and Information Sciences. 34. 10.1016/j.jksuci.2019.06.012.
23. Kamalov, F., Thabtah, F. & Leung, H.H. Feature Selection in Imbalanced Data. Ann. Data. Sci. 10, 1527–1541 (2023). https://doi.org/10.1007/s40745-021-00366-5
24. IDS 2018 Intrusion CSVs (CSE-CIC-IDS2018). https://www.kaggle.com/datasets/solarmainframe/ids-intrusion-csv
25. Aggarwal, P., & Sharma, S. K. (2015). Analysis of KDD dataset attributes - class wise for intru-sion detection. Procedia Computer Science, 57, 842–851. https://doi.org/10.1016/j.procs.2015.07.490
26. NSL-KDD dataset. URL: http://www.unb.ca/research/iscx/dataset/iscx-NSL-KDD-dataset.html.

27. Moustafa, Nour & Slay, Jill. (2015). UNSW-NB15: a comprehensive data set for network intru-sion detection systems (UNSW-NB15 network data set). https://doi.org/10.1109/MilCIS.2015.7348942.
28. Damasevicius, R., Venckauskas, A., Grigaliunas, S., Toldinas, J., Morkevicius, N., Aleliunas, T., & Smuikys, P. (2020). LITNET-2020: An annotated real-world network flow dataset for network intrusion detection. Electronics, 9(5), 800. https://doi.org/10.3390/electronics9050800
29. Emanet S., Karatas Baydogmus G., Demir O. (2023) An ensemble learning based IDS using Voting rule: VEL-IDS. PeerJ Computer Science 9:e1553 https://doi.org/10.7717/peerj-cs.1553

Mohan, Chander. (2019). AN INTRODUCTION TO FUZZY SET THEORY AND FUZZY LOGIC (Second Edition)

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 1 (21), 2023
ISSN 2663 - 4023

Yevhen Chychkarov
doctor of Technical Sciences, professor, professor of the Artificial Intelligence Department
State University of Information and Communication Technologies, Kyiv, Ukraine
ORCID 0000-0002-4362-5129
chychkarovea@gmail.com

Olga Zinchenko
doctor of Technical Sciences, head of the Artificial Intelligence Department
State University of Information and Communication Technologies, Kyiv, Ukraine
ORCID 0000-0002-3973-7814
zinchenkoov@gmail.com

Andriy Bondarchuk
doctor of technical sciences, professor, director of the educational and scientific institute of information technologies
State University of Information and Communication Technologies, Kyiv, Ukraine
ORCID 0000-0001-5124-5102
dekan.it@ukr.net

Liudmyla Aseeva
postgraduate
State University of Information and Communication Technologies, Kyiv, Ukraine
ORCID 0000-0001-5954-4211
aseewal@i.ua

DETECTION OF NETWORK INTRUSIONS USING MACHINE LEARNING ALGORITHMS AND FUZZY LOGIC

Abstract. The study proposed a new method of constructing a set of important features for solving classification problems. This method is based on the idea of using an ensemble of estimators of the importance of features with summarization and the final result of the ensemble with the help of fuzzy logic algorithms. Statistical criteria (chi2, f_classif, correlation coefficient), mean decrease in impurity (MDI), mutual information criterion (mutual_info_classif) were used as estimators of the importance of features. Reducing the number of features on all data sets affects the accuracy of the assessment according to the criterion of the average reduction of classification errors. As long as the group of features in the data set for training contains the first features with the greatest influence, the accuracy of the model is at the initial level, but when at least one of the features with a large impact is excluded from the model, the accuracy of the model is noticeably reduced. The best classification results for all studied data sets were provided by classifiers based on trees or nearest neighbors: DesignTreeClassifier, ExtraTreeClassifier, KNeighborsClassifier. Due to the exclusion of non-essential features from the model, a noticeable increase in the speed of learning is achieved (up to 60-70%). Ensemble learning was used to increase the accuracy of the assessment. The VotingClassifier classifier, built on the basis of algorithms with the maximum learning speed, provided the best learning speed indicators. For future work, the goal is to further improve the proposed IDS model in the direction of improving the selection of classifiers to obtain optimal results, and setting the parameters of the selected classifiers, improving the strategy of generalizing the results of individual classifiers. For the proposed model, the ability to detect individual types of attacks with multi-class prediction is of significant interest.

Keywords: intrusion detection system, machine learning, ensemble learning, classifier, fuzzy logic, cyber attack; cyber defense using machine learning; feature selection algorithms

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Chua, T.-H., & Salam, I. (2023). Evaluation of Machine Learning Algorithms in Network-Based Intrusion Detection Using Progressive Dataset. Symmetry, 15(6), 1251. https://doi.org/10.3390/sym15061251
2. Disha, R. A., & Waheed, S. (2022). Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique. Cybersecurity, 5(1). https://doi.org/10.1186/s42400-021-00103-8
3. Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity, 2(1). https://doi.org/10.1186/s42400-019-0038-7
4. Liao, H.-J., Richard Lin, C.-H., Lin, Y.-C., & Tung, K.-Y. (2013). Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 36(1), 16–24. https://doi.org/10.1016/j.jnca.2012.09.004
5. Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks. IEEE Access, 5, 21954–21961. https://doi.org/10.1109/access.2017.2762418
6. Divekar, A., Parekh, M., Savla, V., Mishra, R., & Shirole, M. (2018). Benchmarking datasets for Anomaly-based Network Intrusion Detection: KDD CUP 99 alternatives. У 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS). IEEE. https://doi.org/10.1109/cccs.2018.8586840
7. Alkasassbeh, M. (2017). An empirical evaluation for the intrusion detection features based on machine learning and feature selection methods. https://doi.org/10.48550/arXiv.1712.09623
8. Catania, C. A., & Garino, C. G. (2012). Automatic network intrusion detection: Current techniques and open issues. Computers & Electrical Engineering, 38(5), 1062–1072. https://doi.org/10.1016/j.compeleceng.2012.05.013
9. Ingre, B., & Yadav, A. (2015). Performance analysis of NSL-KDD dataset using ANN. In: 2015 International Conference on Signal Processing And Communication Engineering Systems (SPACES). IEEE. pp 92–96. https://doi.org/10.1109/spaces.2015.7058223
10. Osanaiye, O., Cai, H., Choo, K.-K. R., Dehghantanha, A., Xu, Z., & Dlodlo, M. (2016). Ensemble-based multi-filter feature selection method for DDoS detection in cloud computing. EURASIP Journal on Wireless Communications and Networking, 2016(1):1-10. https://doi.org/10.1186/s13638-016-0623-3
11. Liu, H., Yan, X., & Wu, Q. (2019). An Improved Pigeon-Inspired Optimisation Algorithm and Its Application in Parameter Inversion. Symmetry, 11(10), 1291. https://doi.org/10.3390/sym11101291
12. Kasongo, S. M., & Sun, Y. (2020). Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset. Journal of Big Data, 7(1). https://doi.org/10.1186/s40537-020-00379-6
13. Wang, X., & Zhou, Y. (2022). Multi-Label Feature Selection with Conditional Mutual Information. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.4153295
14. Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. Evaluating feature importance estimates, 2018. https://doi.org/10.48550/arXiv.1806.10758
15. Rengasamy, D., Rothwell, B. C., & Figueredo, G. P. (2021). Towards a More Reliable Interpretation of Machine Learning Outputs for Safety-Critical Systems Using Feature Importance Fusion. Applied Sciences, 11(24), 11854. https://doi.org/10.3390/app112411854.
16. Souhail et. al., M. (2019). Network Based Intrusion Detection Using the UNSW-NB15 Dataset. International Journal of Computing and Digital Systems, 8(5), 477–487. https://doi.org/10.12785/ijcds/080505
17. Rengasamy, Divish & Mafeni Mase, Jimiama & Rothwell, Benjmain & Torres, Mercedes & Alexander, Morgan & Winkler, David & Figueredo, Grazziela. (2022). Feature Importance in Machine Learning Models: A Fuzzy Information Fusion Approach. Neurocomputing. 511. https://doi.org/10.1016/j.neucom.2022.09.053.
18. Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable Artificial Intelligence

(XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion, 58, 82–115. https://doi.org/10.1016/j.inffus.2019.12.012

19. Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, and Huan Liu. (2017). Feature Selection: A Data Perspective. ACM Comput. Surv. 50, 6, Article 94 (November 2018), 45 pages. https://doi.org/10.1145/3136625
20. Huan Liu and Lei Yu. (2005). Toward Integrating Feature Selection Algorithms for Classification and Clustering. IEEE Trans. on Knowl. and Data Eng. 17, 4 (April 2005), 491–502. https://doi.org/10.1109/TKDE.2005.66
21. Breiman, L. (2017). Classification and Regression Trees (1st ed.). Routledge. https://doi.org/10.1201/9781315139470
22. Khaire, Utkarsh & Dhanalakshmi, R.. (2019). Stability of Feature Selection Algorithm: A Review. Journal of King Saud University - Computer and Information Sciences. 34. 10.1016/j.jksuci.2019.06.012.
23. Kamalov, F., Thabtah, F. & Leung, H.H. Feature Selection in Imbalanced Data. Ann. Data. Sci. 10, 1527–1541 (2023). https://doi.org/10.1007/s40745-021-00366-5
24. IDS 2018 Intrusion CSVs (CSE-CIC-IDS2018). https://www.kaggle.com/datasets/solarmainframe/ids-intrusion-csv
25. Aggarwal, P., & Sharma, S. K. (2015). Analysis of KDD dataset attributes - class wise for intru-sion detection. Procedia Computer Science, 57, 842–851. https://doi.org/10.1016/j.procs.2015.07.490
26. NSL-KDD dataset. URL: http://www.unb.ca/research/iscx/dataset/iscx-NSL-KDD-dataset.html.
27. Moustafa, Nour & Slay, Jill. (2015). UNSW-NB15: a comprehensive data set for network intru-sion detection systems (UNSW-NB15 network data set). https://doi.org/10.1109/MilCIS.2015.7348942.
28. Damasevicius, R., Venckauskas, A., Grigaliunas, S., Toldinas, J., Morkevicius, N., Aleliunas, T., & Smuikys, P. (2020). LITNET-2020: An annotated real-world network flow dataset for network intrusion detection. Electronics, 9(5), 800. https://doi.org/10.3390/electronics9050800
29. Emanet S., Karatas Baydogmus G., Demir O. (2023) An ensemble learning based IDS using Voting rule: VEL-IDS. PeerJ Computer Science 9:e1553 https://doi.org/10.7717/peerj-cs.1553
30. Mohan, Chander. (2019). AN INTRODUCTION TO FUZZY SET THEORY AND FUZZY LOGIC (Second Edition).