УДК 004.056

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ ДЛЯ КЛАСИФІКАЦІЇ ТРАФІКА В МОБІЛЬНИХ МЕРЕЖАХ

А.А. Астраханцев, Л.С. Глоба, А.М. Давидюк, О.В. Сушко
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

Abstract *– The development of mobile networks and implementation of new standards, such as 5G and 6G, in the future will lead to increased traffic volume in the network and new types of traffic creation. Also, new traffic types demand specific service requirements. Currently, existing traffic processing methods are not adapted to such changes, which can impair the Quality of Service. A possible solution for improving the efficiency of information processing is introducing new algorithms for classifying and prioritizing traffic. That is why in this work, the main focus is on analyzing the effectiveness of machine learning algorithms to solve the problem of traffic classification in mobile networks in real-time. The accuracy of classification and performance for the most common machine learning algorithms is analyzed, and the criterion of classification accuracy determines the optimal algorithm to achieve the goal. The results of the comparative analysis showed that the best accuracy could be achieved when using ANN algorithms (the number of latent network layers is 200) and RF. At the same time, the advantages of ANN include high efficiency and reliability of information processing and simple algorithm learning. Also, the RF algorithm is a quick and powerful classification algorithm, but it has shortcomings during the interpretation of the solution and works poorly for small data. In addition, the work assessment of the importance of the dataset fields for classification was evaluated. These improvements can be implemented both on final devices and base stations. They will improve the quality of classification, clustering, and processing of packets, which will generally increase the efficiency of the intellectual mobile network management system. Further development of the topic may be using the studied algorithms to solve the problems of detecting anomalies in traffic to increase the network's security.*

Анотація *– Розвиток мобільних мереж і поява нових стандартів, як-от 5G та в перспективі 6G, призводять до збільшення як обсягів трафіка у мережі, так і до появи нових типів трафіка, зокрема зі специфічними вимогами до обслуговування. Існуючі на цей час методи обробки трафіка не адаптовані до таких змін, що може привести до погіршення якості обслуговування. Можливим шляхом вирішення задачі підвищення ефективності обробки інформації є впровадження нових алгоритмів класифікації та пріоритезації трафіка. У зв`язку з цим у роботі ставиться актуальне завдання аналізу ефективності алгоритмів машинного навчання для вирішення завдання класифікації трафіка в мобільних мережах у режимі реального часу. Для досягнення мети аналізується точність класифікації та швидкодія для найпоширеніших алгоритмів машинного навчання та визначається оптимальний алгоритм за критерієм точності класифікації. Результати порівняльного аналізу показали, що найкращих показників точності можна досягти у разі використання алгоритмів ANN (кількість прихованих шарів мережі дорівнює 200) та RF. При цьому до переваг ANN слід віднести високу оперативність і достовірність обробки інформації, а також простоту у навчанні. В той же час RF хоча і є швидким і потужним алгоритмом класифікації, але він має недоліки під час інтерпретації рішення та неефективний для малих обсягів даних. Крім того, у роботі виконано оцінку важливості полів датасету для класифікації. Вказані вдосконалення можуть бути впровадженні як на кінцевих пристроях, так і на базових станціях, що дозволяє підвищити якість класифікації, кластеризації та обробки пакетів, а також підвищити ефективність інтелектуальної системи управління мобільною мережею загалом. Подальшим розвитком теми може бути застосування досліджуваних алгоритмів для вирішення завдань виявлення аномалій трафіка з метою підвищення захищеності мережі.*

Вступ

Інтенсивний розвиток і впровадження новітніх мереж 5G за останні роки та перспективи впровадження 6G виявили низку нових проблем сучасних мереж [1]. У порівнянні з традиційним зв'язком 3G/4G та трафіком у стільникових мережах, у 5G додається велика кількість напівавтономних та автономних авто, різноманітних smart-виробництв, а відповідно й сенсорів і датчиків. Все це може спричинити серйозні проблеми як у ядрі мережі, так і у мережі радіодоступу (RAN), оскільки призведе до перевантаження та зниження якості обслуговування. Для недопущення пере-

вантажень потрібне вдосконалення існуючих методів попередньої класифікації трафіка та подальший його розподіл на обробку. Для ефективної обробки трафіка у мережах 5G/6G є ключова особливість – мережні зрізи (network slicing) [2], які дозволяють розподіляти ресурси системи залежно від типу додатку і обробляти кожен зріз (слайс) окремо (рис. 1). Для ефективної роботи цієї особливості також дуже важлива попередня класифікація та розмітка (маркування) трафіка.

Класифікація мережного трафіка дозволяє у подальшому організувати його диференційоване обслуговування відповідно до вимог щодо рівня якості обслуговування (QoS) [3], що дозволяє виділити мережні ресурси для забезпечення оптимальних QoS-показників для різних класів трафіка. Наприклад, мережний трафік з високим пріоритетом або конкретні критерії, що відповідають трафіка, можуть бути виділені для спеціальної обробки, а отже, допомогти досягти пікових продуктивностей як додатків, так і мережі загалом.

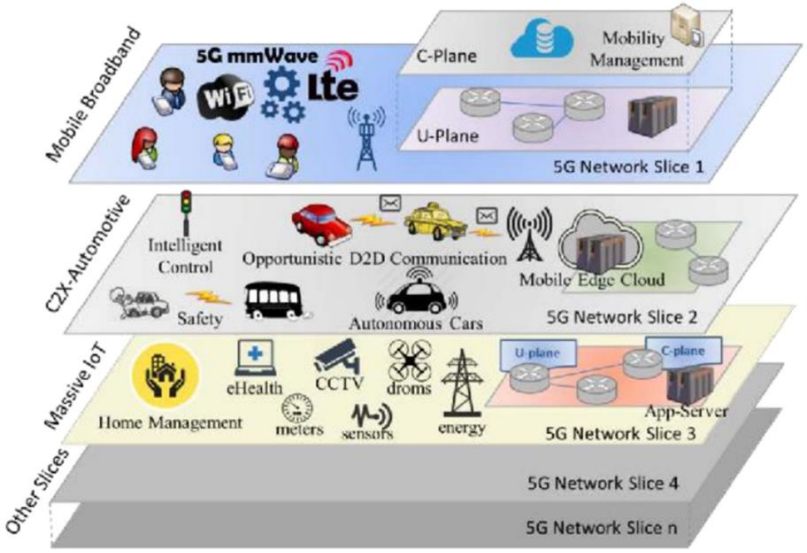

Рис. 1. Мережні слайси для мережі 5G [1]

Обробка трафіка з урахуванням якості обслуговування може включати більш швидку переадресацію за допомогою проміжних маршрутизаторів і комутаторів або зменшення ймовірності відкидання пакетів трафіка через відсутність ресурсів у проміжних вузлах.

Мета роботи: аналіз ефективності алгоритмів машинного навчання для вирішення завдання класифікації трафіка у мобільних мережах 5G/6G за критеріями якості класифікації та швидкодії. За підсумками аналізу мають бути сформовані рекомендації по застосуванню алгоритмів машинного навчання та визначені їх оптимальні параметри.

I. Огляд існуючих рішень

Класифікація мережного трафіка може бути здійснена на основі використання інформації різних рівнів моделі OSI (Open Systems Interconnection). На фізичному рівні можна виконувати аналіз і класифікацію на основі бітових послідовностей та

обсягу трафіка [4]. На більш верхніх рівнях для цього можуть бути використані номерів портів, вміст пакета, ідентифікатори потоку, заголовки пакетів. Водночас характеристики мережного трафіка на кожному з рівнів відрізняються. Наприклад, на рівні потоку пакетів мережний трафік характеризується розміром пакета та часовим інтервалом між пакетами. Аналіз на рівні бітової послідовності переважно стосується таких характеристик, як інтенсивність передачі і пропускна здатність каналу. На рівні потоку пакетів розглядається також процедура прибуття IP-пакетів, тобто їх затримки і втрати.

В [5] вказується висока актуальність теми і показується можливе подальше використання результатів класифікації для вирішення завдання підвищення інформаційної безпеки в розподіленому обчислювальному середовищі. Також пропонується використання алгоритмів на основі «найближчих сусідів» (KNN).

В [6, 7] вказується що існують різноманітні методи класифікації трафіка, які базуються на портах пакетів, урахуванні навантаження та на машинному навчанні. У цих роботах робиться акцент на дослідженні таких алгоритмів машинного навчання, як Support Vector Machine (SVM), decision tree, Naive Bayes та Bayes Net.

У багатьох роботах, наприклад [8-12] пропонуються для використання такі алгоритми, як random forest (RF), KNN, ANN і SVM. Тому саме ці алгоритми були обрані для дослідження в даній роботі.

Крім алгоритмів машинного навчання для класифікації трафіка, можуть застосовуватися методи «глибокого огляду пакетів» (Deep Packet Inspection, DPI). Deep Packet Inspection – це найсучасніша технологія для класифікації трафіка, оскільки вона є найбільш точною методикою [13]. Тому часто найпопулярніші продукти, як комерційні, так і відкриті, покладаються на DPI під час класифікації трафіка. Однак фактична ефективність DPI все ще невизначена, оскільки обмежена кількість публічних наборів даних обмежує порівняння та відтворювання результатів [13].

У даній роботі обрані для дослідження саме алгоритми на основі машинного навчання. Інші методи будуть проаналізовані та досліджені в наступних публікаціях.

II. Алгоритми класифікації трафіка

Класифікатор трафіка дозволяє ідентифікувати шаблон трафіка, який відповідає одному з апріорно відомих класів. Для класифікації, як наведено в [4-13], можуть використовуватися алгоритми штучних нейронних мереж (artificial neural networks, ANN), *k*-найближчих сусідів (*k*-nearest neighbor, KNN), «випадкового лісу» (Random Forest, RF). Розглянемо їх принципи роботи, переваги та недоліки більш детально.

Алгоритм штучних нейронних мереж (ANN). Штучна нейронна мережа (ШНМ) – це система з'єднаних і взаємодіючих між собою штучних нейронів (простих процесорів) [14, 15]. Кожен штучний нейрон має справу з сигналами, які він періодично отримує, і сигналами, які він періодично посилає іншим нейронам. Штучна нейронна мережа працює наступним чином: на входи нейронів надходять сигнали, які підсумовуються. При цьому враховується синаптична вага, тобто значимість кожного з

входів. Далі вхідні сигнали одних нейронів надходять на входи інших нейронів. Вага кожного такого зв'язку може бути позитивною (збуджуючі зв'язки) або негативною (гальмівні зв'язки). Вони визначають обчислення нейронної мережі, та, відповідно, її пам'ять та поведінку.

Штучна нейронна мережа складається з трьох компонентів (рис. 2):

- вхідний шар;
- приховані (обчислювальні) шари;
- вихідний шар.

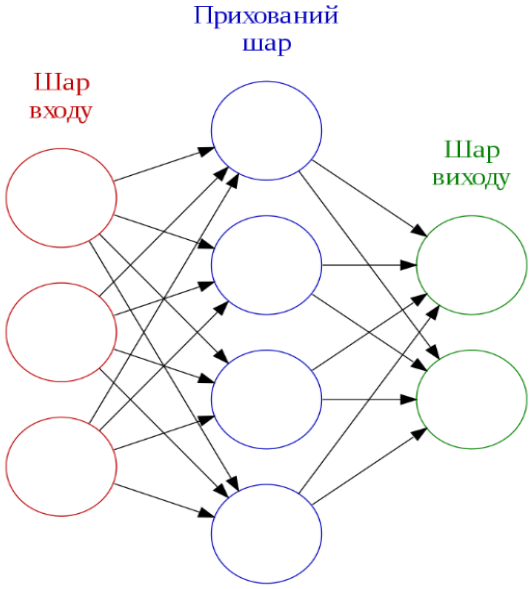

Рис. 2. Загальний вигляд штучної нейронної мережі

Для отримання результату – виконання класифікації трафіка необхідно попередньо виконати навчання нейронної мережі. Під навчанням будемо розуміти пошук набору вагових коефіцієнтів, які при проходженні через суматор дозволять отримати потрібний сигнал. Навчання таких нейромереж відбувається у два етапи: пряме поширення помилки і зворотне поширення помилки. Під час прямого поширення помилки робиться прогноз відповіді. При зворотному поширенні помилка між фактичною відповіддю та передбаченою мінімізується.

Переваги штучних нейронних мереж:

1. Висока надійність роботи. Інформація в ШНМ кодується і запам'ятовується не в окремих елементах пам'яті, а в розподілі зв'язків між нейронами та в їх силі, тому стан кожного окремого нейрона визначається станом багатьох інших нейронів, пов'язаних з ним. Тому втрата одного або декількох зв'язків не має істотного впливу на результат роботи системи взагалі, що забезпечує її високу надійність.

2. Висока «природна» завадостійкість і функціональна надійність стосуються як спотворених (зашумлених) потоків інформації, так і відмов окремих нейронів. Цим забезпечуються висока оперативність і достовірність обробки інформації, а просте

донавчання і перенавчання мереж дозволяють при зміні зовнішніх чинників своєчасно здійснювати перехід на новий рівень вирішуваних завдань.

Алгоритм k-найближчих сусідів (KNN). Це простий непараметричний класифікаційний алгоритм, де для класифікації об'єктів у межах простору властивостей використовуються відстані (зазвичай евклідові), пораховані до усіх інших об'єктів [16, 17]. З метою згладжування шумового впливу викидів алгоритм класифікує об’єкти шляхом голосування за k найближчими сусідами. Кожен із сусідів $y^{(j)}$, $(j=\overline{1,k})$ голосує за віднесення реалізації образу до свого класу. Алгоритм відносить реалізацію, що розпізнається, до того класу, який набере найбільше число голосів. Оптимальне значення параметра k визначається за критерієм ковзного контролю з виключенням об’єктів по одному. Для кожного об’єкта перевіряється, чи правильно він класифікується за своїми k найближчими сусідами (рис. 3). Тестовий зразок (зелене коло на рис. 3) повинен бути класифікований як синій квадрат (клас 1) або як червоний трикутник (клас 2). Якщо k = 3, то тестовий зразок буде віднесено до 2-го класу, якщо k = 5, то він буде класифікований як клас 1.

Для віднесення «сусідами» тестового зразку до свого класу можуть використовуватися різноманітні міри близькості. Найбільш поширеними є манхеттенська відстань та ступенева відстань, окремим випадком якої є Евклідова відстань.

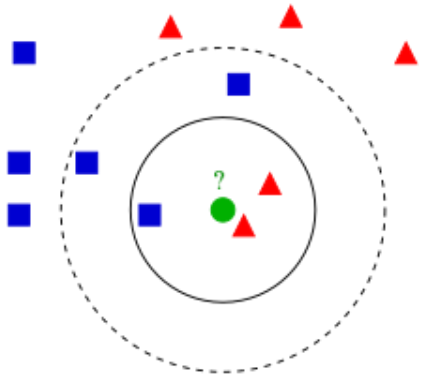

Рис. 3. Приклад класифікації k найближчих сусідів

Манхеттенська відстань – відстань між двома точками, яка дорівнює сумі модулів різниць їх координат

$$d(x,y)=\sum_{i=1}^{N}\left|x_i-y_i\right|. \qquad (1)$$

У більшості випадків ця міра відстані призводить до таких результатів, як і Евклідова відстань. Однак для цієї міри вплив окремих великих різниць (викидів) зменшується.

Ступенева відстань застосовується у випадках, коли необхідно збільшити або зменшити вагу, що відноситься до розмірності, для якої об’єкти суттєво відрізняються. Ступенева відстань розраховується за формулою

$$d(x,y) = r\sqrt{\sum_{i=1}^{N}(x_i - y_i)^p}\ , \qquad (2)$$

де r і p – параметри, визначені користувачем.

Параметр r відповідає за поступове зважування різниць по окремим координатам, а параметр p – за прогресивне зважування великих відстаней між об'єктами. Якщо r і p дорівнюють 2, то ця відстань співпадає з відстанню Евкліда. Міра близькості підбирається індивідуально для конкретних типів даних.

Алгоритм KNN має як переваги, так і недоліки. До переваг слід віднести наступне:

- алгоритм стійкий до аномальних викидів, тому що ймовірність влучення такого запису в число k-найближчих сусідів мала. Якщо ж це відбулося, то вплив на голосування, особливо зважене, при k>2, швидше за все, буде незначним, і, отже, малим буде і вплив на підсумок класифікації;
- програмна реалізація алгоритму відносно проста;
- результат роботи алгоритму легко піддається інтерпретації;
- можливість модифікації алгоритму, використання найбільш придатних функцій сполучення і метрик дозволяє налаштовувати алгоритм під конкретну задачу.

До недоліків алгоритму відносять таке:

- набір даних, використаний для алгоритму, повинен бути репрезентативним;
- модель не можна "відокремити" від даних: для класифікації нового прикладу потрібно використовувати всі приклади. Ця особливість сильно обмежує використання алгоритму.

Алгоритм «випадкового лісу» (Random Forest, RF). RF базується на деревах рішень і може бути використаний як для класифікації, так і для регресійних завдань [18, 19]. У машинному навчанні дерева рішень є алгоритмом створення моделей прогнозування. Їх називають деревами прийняття рішень, оскільки передбачення слідує за кількома гілками рішення "якщо…, тоді…", поділеним на гілки дерева. Цей поділ можна розглядати як функцію в машинному навчанні. Рішення будуть прийматися, поки не відбудеться перехід до наступної гілки і не повториться той самий процес прийняття рішень, поки не буде більше гілок. Ця кінцева точка називається листом, а в деревах рішень – кінцевий результат: прогнозований клас або значення. У кожній гілці є порогові значення, які найкраще розділяють дані, що залишилися. RF робить прогнози шляхом комбінування результатів з багатьох окремих дерев рішень.

Для налаштування навчання алгоритму використовують гіперпараметри. Гіперпараметри – це аргументи, які можна встановити перед тренуванням і які визначають, як проводиться навчання. Основними гіперпараметрами у Random Forest є:

- кількість дерев рішень, які необхідно об'єднати;
- максимальна глибина дерев;
- максимальна кількість ознак, що розглядаються при кожному розбитті;
- тип навчання класифікаторів (паралельне чи послідовне).

Переваги Random Forests полягають у тому, що він є відносно швидким і потужним алгоритмом навчання, класифікації та регресії. Розрахунки можуть бути паралелізовані та добре виконуються при багатьох задачах, навіть при малих наборах даних, а вихідні дані повертають імовірності прогнозування.

Недоліки Random Forests полягають у тому, що немає можливості інтерпретувати рішення, прийняті моделлю, тому що вони занадто складні. RF також дещо схильні до перенавчання, і вони зазвичай неточно прогнозують недостатньо представлені класи у незбалансованих наборах даних.

Для дослідження ефективності застосування алгоритмів машинного навчання під час розв'язання задач класифікації трафіка використаний розмічений датасет з [20]. Враховуючи, що більшість наборів даних класифікації мережного трафіка спрямовані лише на ідентифікацію типу додатку, який використовує потік IP (www, dns, ftp, p2p, telnet тощо), цей набір даних іде на крок далі, дозволяючи виявити конкретні додатки, такі як Facebook, YouTube, Instagram тощо, зі статистики потоку IP.

Відповідно до опису датасета він був зібраний в університеті Universidad Del Cauca (Колумбія), містить більш ніж 3,5 млн пакетів, отриманих за 6 діб від 75 додатків, наведених нижче. Зібрані дані структуровані у вигляді CSV файлу. Важливою особливістю даного датасету окрім великої кількості мережних додатків є значна кількість полів пакетів, які використовуються як ознаки для навчання моделі та класифікації пакетів. Фрагмент датасету з даними (перші 5 ознак) наведено в табл. 1.

Таблиця 1. Фрагмент датасету

Flow ID	Source IP	Source Port	Destination IP	Destination Port	Flow Duration
172.19.1.46-10.200.7.7-52422-3128-6	172.19.1.46	52422	10.200.7.7	3128	45523
172.19.1.46-10.200.7.7-52422-3128-6	10.200.7.7	3128	172.19.1.46	52422	1
10.200.7.217-50.31.185.39-38848-80-6	50.31.185.39	80	10.200.7.217	38848	1
10.200.7.217-50.31.185.39-38848-80-6	50.31.185.39	80	10.200.7.217	38848	217
192.168.72.43-10.200.7.7-55961-3128-6	192.168.72.43	55961	10.200.7.7	3128	78068
172.19.1.56-10.200.7.6-50004-3128-6	10.200.7.6	3128	172.19.1.56	50004	105069

III. Аналіз ознак для класифікації трафіка

Розглянемо поля пакетів, наведені в датасеті, що можуть бути використані як ознаки для навчання моделі та класифікації трафіка. Короткий перелік полів наведено у табл. 2.

Таблиця 2. Перелік полів для класифікації трафіка

Flow ID	Flow Duration	Flow bytes/s	Average Packet Size
Source IP	Total Fwd Packets	Flow packets/s	Avg Fwd Segment Size
Source Port	Total Backward Packets	Average Packet Size	Avg Bwd Segment Size
Destination IP	Total Length of Fwd Packets	Flags (x 8)	Fwd Header Length.
Destination Port	Total Length of Bwd Packets	Packet Length Mean	Down Up Ratio
Protocol	Fwd Packet Length Max	Bwd Packet Length Max	Label
Timestamp	Fwd Packet Length Min	Bwd Packet Length Min	App name

Як видно з табл. 2, присутня достатня кількість полів, які є основними і обов'язковими для виконання класифікації з високою точністю, але є і велика кількість полів, вплив яких важко оцінити на цьому етапі. Варто підкреслити, що з одного боку збільшення кількості полів (ознак) для класифікації дозволяє підвищити точність класифікації, з іншого боку зростання кількості ознак підвищує її складність, тому після визначення найкращих алгоритмів класифікації необхідно буде виконати оптимізацію ознак класифікації для формування мінімально достатнього набору полів (ознак), що буде забезпечувати задану точність класифікації.

Для оцінки ефективності алгоритмів машинного навчання будуть використовуватися різні метрики [8]: точність (accuracy), чіткість (precision), відкликання (recall) та *F*1-метрика.

Під *Accuracy* будемо розуміти відношення правильно класифікованих зразків (пакетів) потоку трафіка до загальної кількості зразків:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{3}$$

де *TP* (True Positive) – це число пакетів, які були правильно класифіковані до певного додатку/сервісу; *TN* (True Negative) – це кількість пакетів, які були правильно класифіковані як такі, що не відповідають додатку/сервісу; *FP* (False Positive) – це кількість пакетів, що була неправильно віднесена до додатку/сервісу; *FN* (False Negative) – це кількість пакетів неправильно розпізнаних, як такі, що не відносяться до додатку/сервісу.

В роботі буде використовуватися середнє значення в десятикратній перехресній перевірці для вимірювання точності та підвищення надійності результатів.

У деяких випадках, коли набір даних має додаток/сервіс, що представляє більшість значень вибірки, значення балів точності може неточно відображати продуктивність моделі класифікатора. Щоб уникнути цієї проблеми, також будуть використані інші метрики оцінки ефективності, як-от чіткість та *F*1-метрика.

Precision (чіткість) – це міра співвідношення позитивних, правильно прогнозованих пакетів у трафіку до загальної кількості позитивних прогнозів класифікації:

$$Precision = \frac{TP}{TP + FP}. \quad (4)$$

Recall (відкликання) вимірює співвідношення фактичних позитивних, правильно прогнозованих пакетів у трафіку:

$$Recall = \frac{TP}{TP + FN}. \quad (5)$$

Метрика *F*1 відображає середнє значення чіткості та відкликань:

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}. \quad (6)$$

III. Результати досліджень

На першому етапі досліджень був відфільтрований список додатків, які не можуть бути успішно класифіковані на основі даних досліджуваного датасету. Як критерій фільтрації використовувалася кількість наявних пакетів. Були відкинуті 25 додатків з найменшою кількістю пакетів (менше ніж 500 пакетів в датасеті). Прикладами таких додатків та веб-сервісів є 'H323', 'ORACLE', 'TEAMSPEAK', 'BGP', 'BITTORRENT', 'OPENSIGNAL', 'MAIL_IMAPS', 'IP_OSPF', 'RADIUS', 'OPENVPN', 'SNMP', 'STARCRAFT', 'QQ', '99TAXI'. Це дозволило підготувати збалансований датасет (рис. 4). Як видно з рис. 4 найбільш представленими в датасеті є додатки типу browsing (Google, http), до яких відносяться більше ніж 600000 пакетів, також широко представлені додатки класу media (Youtube) та пошти (Gmail).

Використовуючи збалансований датасет для алгоритму штучних нейронних мереж (ANN), була виконана оцінка максимально досяжної точності та її залежність від шкидкодії (рис. 5 та 6)

Як видно з наведених графіків (рис. 5 та 6) зі збільшенням кількості шарів підвищується точність, але й нелінійно збільшується швидкодія. Тому для пришвидшення обробки або забезпечення обчислень в реальному часі без затримки варто обмежити кількість шарів на рівні 180-220 шарів (точність відповідно 0,987-0,99). Найкращим з боку точності є значення 512 шарів.

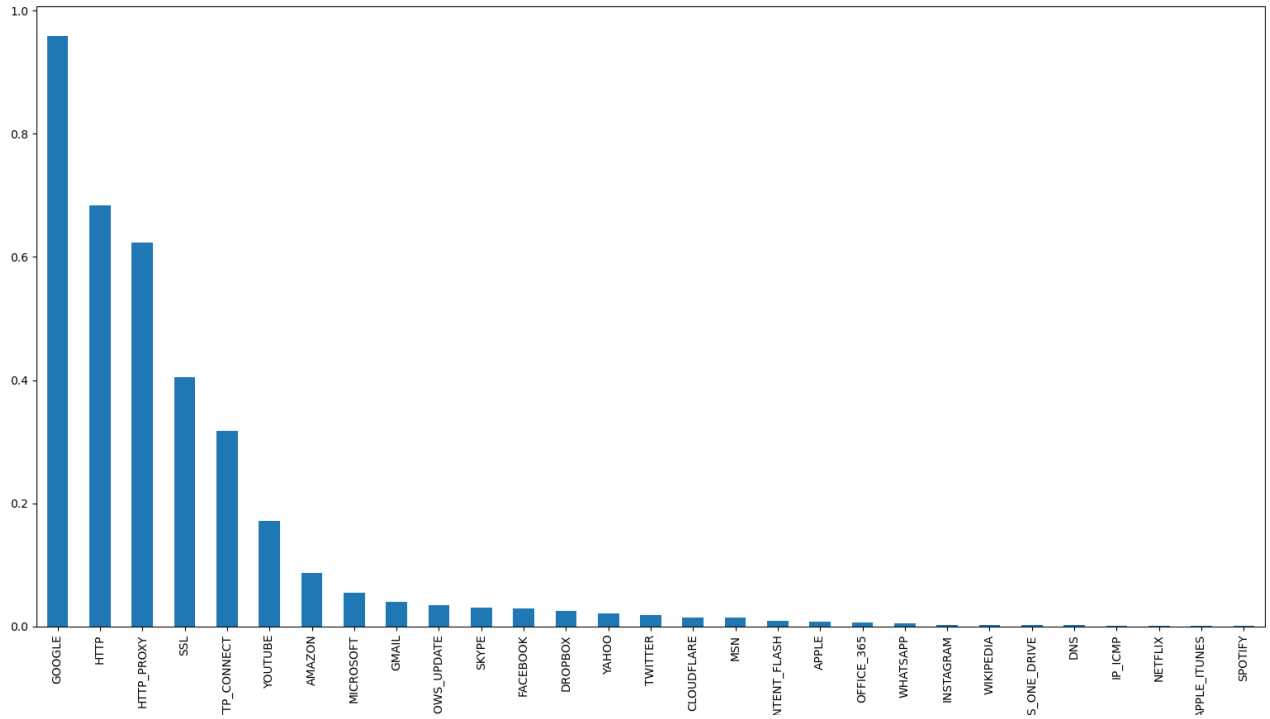

Рис. 4. Кількість пакетів для кожного з додатків в збалансованому датасеті (по осі ординат наведена Occurrence x 10^6 – кількість пакетів в датасеті)

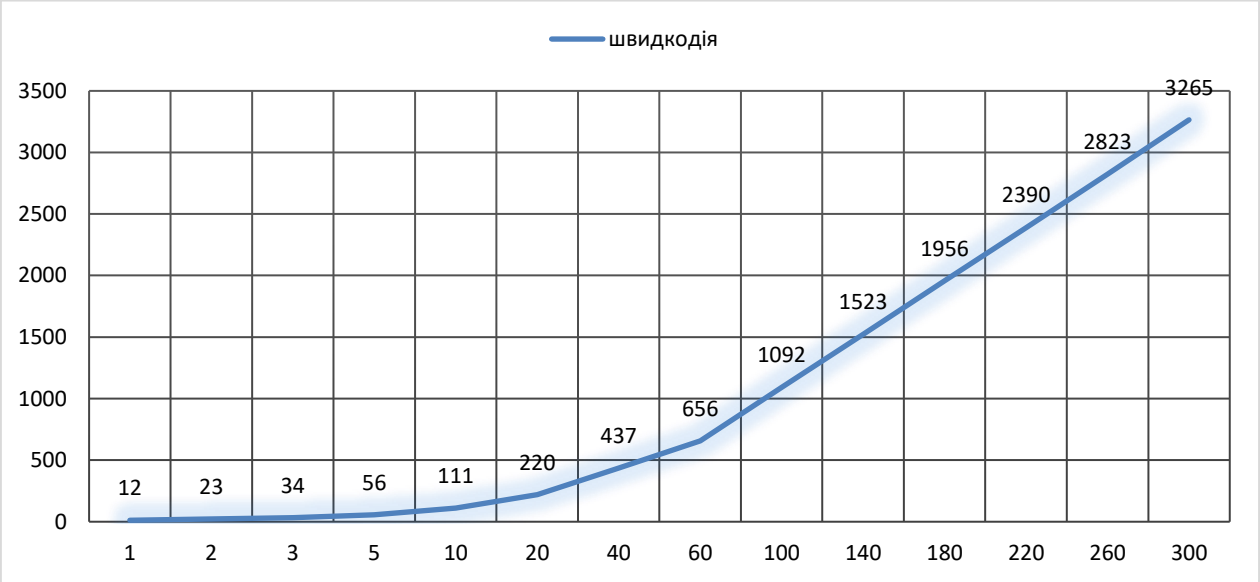

Рис. 5. Залежність швидкодії класифікації для збалансованого датасету від кількості шарів в мережі (1-300)

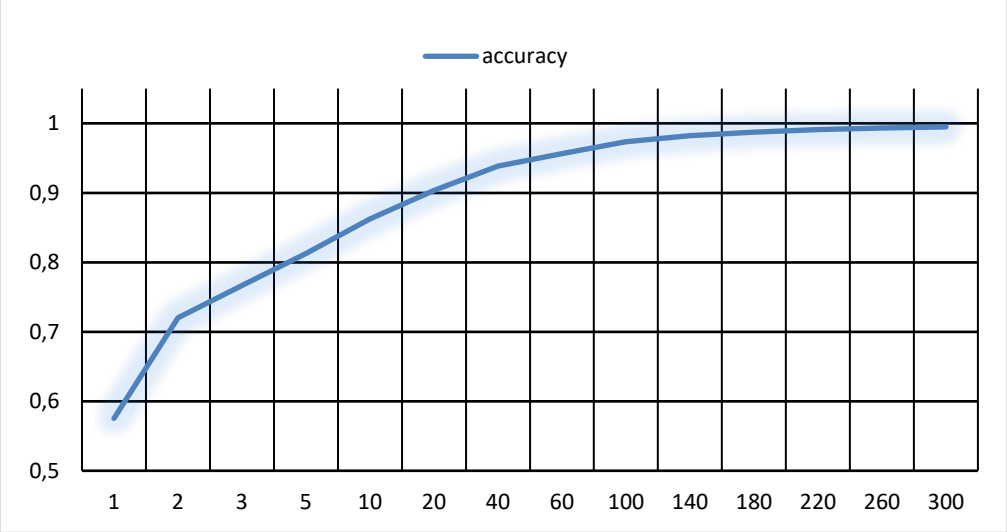

Рис. 6. Залежність точності класифікації від кількості шарів в мережі ANN (1-300)

Для алгоритму *k*-найближчих сусідів (KNN) у роботі виконано аналіз впливу кількості сусідів на точність (рис. 7) та визначено, що найвища точність класифікації забезпечується при використанні манхеттенської відстані і при $k = 3$. В цьому випадку можна досягти точності класифікації на рівні 0,93. Як видно з рис. 7, точність незначно змінюється при збільшенні числа сусідів, і для даного датасету збільшення кількості сусідів не покращує результати класифікації.

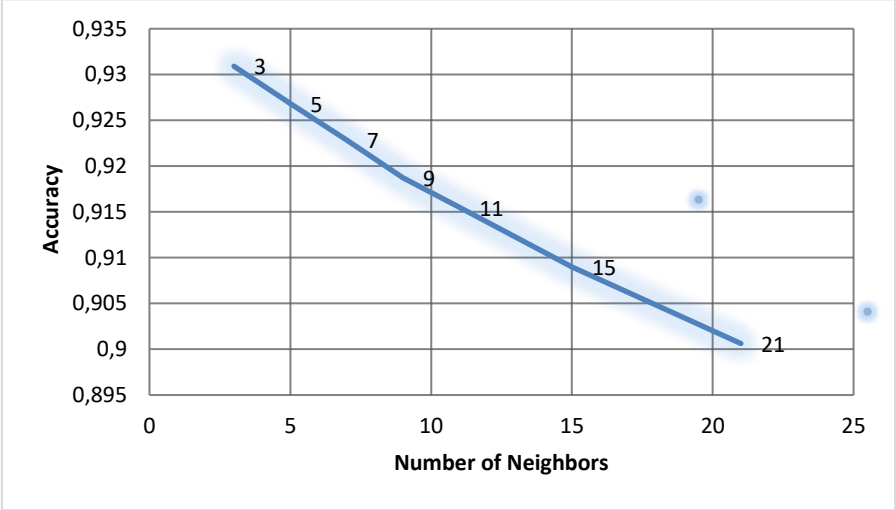

Рис. 7. Залежність точності класифікації від кількості сусідів для алгоритму KNN

Також для збалансованого датасету була виконана оцінка найкращих параметрів моделі та точності класифікації у разі використання алгоритму «випадкового лісу» (RF). Для алгоритму RF оптимальними за критерієм точності розпізнавання були такі параметри:

- кількість дерев рішень, які необхідно об'єднати (n_estimators), дорівнює 50;
- максимальна глибина дерев складала 40;

- максимальна кількість ознак, що розглядаються при кожному розбитті, становила log2.

За наведених вище умов можна досягти точності класифікації на рівні 0,993. Порівняльна характеристика результатів оцінювання точності класифікації алгоритмом RF для різних значень кількості дерев наведена на рис. 8.

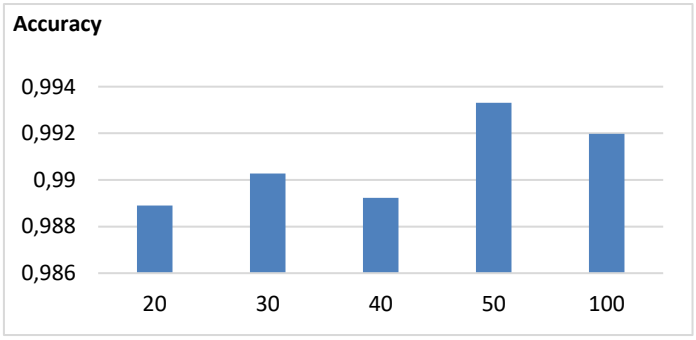

Рис. 8. Залежність точності класифікації від кількості дерев рішень для алгоритму RF

Точність класифікації в цьому випадку залежить від того, які поля пакетів (інформація про трафік) присутні в датасеті, і від важливості цих полів для розпізнавання. Порівняння важливості полів для алгоритму RF наведено на рис. 9.

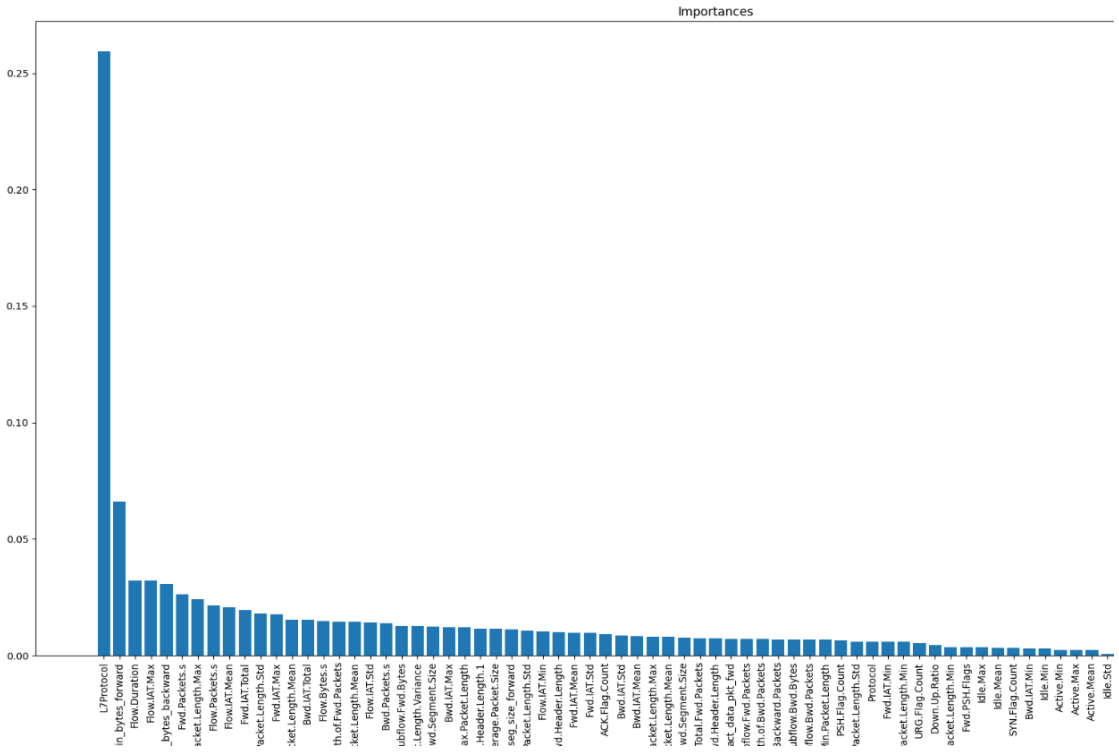

Рис. 9. Важливість полів датасету для проведення класифікації (на прикладі роботи RF)

Як видно з рис. 9, найсильніший вплив на класифікацію мають інформація щодо протоколів верхнього (7-го) рівня OSI, мінімальний розмір пакета, що передається, та тривалість сесії.

Підсумкові результати порівняння точності класифікації під час використання різних алгоритмів, наведено в табл. 3.

Таблиця 3. Порівняння ефективності класифікації для алгоритмів машинного навчання

Алгоритм	Accuracy	Precision	Recall	*F*1-score
ANN	0,9908	0,9920	0,9908	0,9908
KNN	0,9329	0,9329	0,9329	0,9329
RF	0,9933	0,9933	0,9933	0,9933

Відповідно до табл. 3 загальні результати класифікації достатньо високі, але ці значення є усередненими, і різні додатки класифікуються з різним ступенем точності. У роботі також була проведена оцінка точності класифікації залежно від типу додатку. В табл. 4 наведено результати для додатків з найкращим та найгіршим рівнем розпізнавання.

Таблиця 4. Результати класифікації додатків алгоритмом KNN

Додаток (протокол) з найбільшою точністю	Accuracy	*Додаток (протокол)* з найменшою точністю	Accuracy
IP_ICMP	1	CITRIX_ONLINE	0,83
NTP	1	UPNP	0,82
TEAMVIEWER	1	GMAIL	0,79
DNS	1	WAZE	0,77
SSH	1	TWITTER	0,77
FTP_CONTROL	1	SKYPE	0,77

Висновки

В даній роботі вирішено актуальну задачу підвищення ефективності системи мобільного зв'язку за рахунок застосування алгоритмів машинного навчання для класифікації трафіка. Результати порівняльного аналізу показали, що найкращих показників точності можна досягти у разі використання алгоритмів ANN та RF. Тоді як до переваг ANN слід віднести високу оперативність і достовірність обробки інформації, а також просте донавчання. Водночас алгоритм RF хоча є швидким і потужним алгоритмом класифікації, але має недоліки під час інтерпретації рішення і погано відпрацьовує для малих обсягів даних.

У зв'язку з цим можна зробити висновок, що найбільш перспективним є застосування алгоритмів на основі ANN.

Під час порівняльного аналізу для всіх алгоритмів за допомогою додатку на мові Python були визначені найкращі параметри роботи. Зазначена точність алгоритму ANN досягається при кількості прихованих шарів мережі, що дорівнює 200.

Також результати досліджень показали, що різні додатки мають різну точність розпізнавання (табл. 4), яка не залежить від загальної кількості пакетів у датасеті (за умови, що початково датасет був збалансований відкиданням додатків з кількістю пакетів менше 500). Найкращими для розпізнавання виявилися пакети управління FTP, пакети DNS, SSH, IP_ICMP – тобто значна доля службового трафіка. Додатки користувача, такі як пошта (Gmail), комунікації (Twitter, Skype) виявилися найскладнішими по віднесенню до свого класу.

Наукова новизна роботи полягає у визначенні оптимальних за критерієм точності параметрів алгоритмів машинного навчання для розв'язання задачі класифікації трафіка в мережах мобільного зв'язку 5-го та 6-го поколінь. Також до наукової новизни слід віднести оцінку важливості параметрів (полів) датасету для класифікації. Запропоновані алгоритми та параметри є першим етапом багатокрокової обробки пакетів в мережі, що разом з кластеризацією, слайсінгом та розподіленою обробкою дозволять підвищити ефективність системи мобільного зв'язку загалом.

Практична значущість роботи полягає в можливості використання вказаних алгоритмів із запропонованими параметрами для підвищення ефективності класифікації пакетів у мережі мобільного зв'язку 5-го та 6-го поколінь.

Список літератури

1. *Sanjay, N.* (2020). 5G Technology: An Overview. Medium. URL: https://medium.com/@nagasanjayvijayan/5g-technology-an-overview-275cfb61cfd3 (last accessed 17.08.2022)

2. How Network Slicing works and why it is key to 5G. Telefónica. URL: https://www.telefonica.com/en/communication-room/blog/how-network-slicing-works-and-why-it-is-key-to-5g/ (last accessed 17.08.2022)

3. QoS: Classification Configuration Guide, Cisco IOS Release 15M&T-Classifying Network Traffic, 2017. 16p. URL: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_classn/configuration/15-mt/qos-classn-15-mt-book/qos-classn-ntwk-trfc.pdf

4. *Заборовский, В.С.* (2010), Анализ трафика в сетях коммутации пакетов, СПБб.: СПбГПУ, 90с.

5. *Довбиш, А.С.* (2016), Звіт про науково-дослідну роботу «Інтелектуальна система керування навантаженням і ресурсами розподіленого обчислювального середовища з підвищеною інформаційною безпекою», СумДУ, 166 с.

6. *Shafiq, M., Yu, X., Laghari, A. A., Yao, L., Karn, N. K., Abdessamia, F.* (2016), "Network Traffic Classification techniques and comparative analysis using Machine Learning algorithms," Proceedings of the 2016 2nd IEEE International Conference on Computer and Communications (ICCC), pp. 2451-2455. DOI: https://doi.org/10.1109/CompComm.2016.7925139

7. *Raikar, M.M., Meena, S.M., Mulla, M.M., Shetti, N.S., Karanandi, M.* (2020), "Data traffic classification in software defined networks (SDN) using supervised-learning", Procedia Computer Science, No. 171, P. 2750-2759. DOI: https://doi.org/10.1016/j.procs.2020.04.299

8. *AlZoman, R.M.; Alenazi, M.J.F.* (2020), "A Comparative Study of Traffic Classification Techniques for Smart City Networks", Sensors, No. 21, 4677, P. 1-17. **DOI**: https://doi.org/10.3390/s21144677

9. *Salman, O., Elhajj, I.H., Kayssi, A., Chehab, A.* (2020), "A review on machine learning–based approaches for Internet traffic classification", Annals of Telecommunications, No. 75(11), P. 673–710. **DOI**: https://doi.org/10.1007/s12243-020-00770-7

10. *Alqudah, N.; Yaseen, Q.* (2020), "Machine Learning for Traffic Analysis: A Review", Procedia Computer Science, No. 170, P. 911-916. **DOI**: https://doi.org/10.1016/j.procs.2020.03.111

11. *Xie, J., Yu, F.R., Huang, T., Xie, R., Liu, J., Wang, C., Liu, Y.* (2018), "A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges", IEEE Communications Surveys & Tutorials, No. 21(1), P. 393-430. **DOI**: https://doi.org/10.1109/COMST.2018.2866942

12. *Aureli, D., Cianfrani, A., Diamanti, A., Vilchez, J.M.S., Secci, S.* (2020), "Going beyond diffserv in ip traffic classification", Proceedings of the NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium, P. 1-6. **DOI**: https://doi.org/10.1109/NOMS47738.2020.9110430

13. *Bujlow, T., Carela-Español, V., Barlet-Ros, P.* (2015), "Independent comparison of popular DPI tools for traffic classification", Computer Networks, No. 76, P.75-89. **DOI**: https://doi.org/10.1016/j.comnet.2014.11.001

14. *Gurney, K.* (1997), An introduction to neural networks, CRC Press, 248 p.

15. *Bhadeshia, H. K. D. H.* (1999), "Neural Networks in Materials Science", ISIJ International, No. 39(10), P. 966-979. **DOI**: https://doi.org/10.2355/isijinternational.39.966

16. Холл, П., Парк, Б., Семворт, Р.Дж. (2008), "Вибір порядку сусідів у класифікації найближчих сусідів", No. 5, С. 2135–2152.

17. *Altman, N.S.* (1992), "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression", The American Statistician, No. 46(3), P. 175-185. **DOI**: https://doi.org/10.1080/00031305.1992.10475879

18. *Breiman, L.* (2001), Random Forests. Machine Learning, No. 45. P. 5-32. **DOI**: http://dx.doi.org/10.1023/A:1010933404324

19. *Hastie, T, Tibshirani, R., Friedman, J.* (2009), Chapter 15. Random Forests. The Elements of Statistical Learning, P. 587-623.

20. IP Network Traffic Flows Labeled with 75 Apps URL: https://www.kaggle.com/jsrojas/ip-network-traffic-flows-labeled-with-87-apps (last accessed 17.08.2022)