

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістра

(освітній ступінь (освітньо-кваліфікаційний рівень))

на тему: **Методи та засоби прогнозування трафіку комп'ютерних мереж
на основі нейромережових моделей**

Виконав: студент (ка) 6 курсу, групи СІМ-61
спеціальності 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

	(підпис)	Бутинець В.Ю. (прізвище та ініціали)
Керівник	(підпис)	Луцків А.М. (прізвище та ініціали)
Нормоконтроль	(підпис)	Луцик Н.С. (прізвище та ініціали)
Завідувач кафедри	(підпис)	Осухівська Г.М. (прізвище та ініціали)
Рецензент	(підпис)	(прізвище та ініціали)

Тернопіль
2020

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра комп'ютерних систем та мереж

ЗАТВЕРДЖУЮ

Завідувач кафедри Осухівська Г.М.

«_____» _____ 2020 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня магістр
(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

студенту Бутинцю Віталію Юрійовичу
(прізвище, ім'я, по-батькові)

1. Тема проекту (роботи) Методи та засоби прогнозування трафіку комп'ютерних мереж на основі нейромережесевих моделей

Керівник проекту (роботи) Луцків Андрій Мирославович, к.т.н., доц.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «28» вересня 2020 року №4/7-687

2. Термін подання студентом завершеної роботи _____

3. Вихідні дані до роботи Тип комп'ютерних мереж, тип трафіку, вихідний набір даних, математичні методи і моделі прогнозування, мова програмування Python

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Аналіз сучасних досліджень у сфері у галузі прогнозування трафіку комп'ютерних мереж. 2. Побудова моделі та розробка алгоритму прогнозування трафіку у комп'ютерних мережах. 3. Реалізація запропонованих моделей та методу прогнозування трафіку у комп'ютерних мережах. 4. Охорона праці та безпека в надзвичайних ситуаціях. Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1. Актуальність і мета дослідження. 2. Задачі дослідження, об'єкт і предмет, наукова новизна і практична цінність дослідження. 3. Методи класифікації трафіку і його прогнозування. 4. Прогнозування трафіку у комп'ютерних мережах на основі моделі ARMA/ARIMA. 5. Результати прогнозування трафіку у комп'ютерних мережах на основі моделі ARMA/ARIMA. 6. Прогнозування трафіку у комп'ютерних мережах на основі deep learning моделей. 7. Архітектура нейронної мережі. 8. Use case діаграми та архітектура програмного засобу для прогнозування трафіку у комп'ютерних мережах. 9. Результати апробації моделі. 10. Висновки

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Охорона праці та безпека в надзвичайних ситуаціях</i>	<i>Осухівська Г.М.</i>		
	<i>Стадник І.Я.</i>		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1.	<i>Аналіз сучасних досліджень у сфері у галузі прогнозування трафіку комп'ютерних мереж</i>	<i>30.09.2020-13.10.2020</i>	<i>виконано</i>
2.	<i>Побудова моделі та розробка алгоритму прогнозування трафіку у комп'ютерних мережах</i>	<i>14.10.2020 – 28.10.2020</i>	<i>виконано</i>
3.	<i>Реалізація запропонованих моделей та методу прогнозування трафіку у комп'ютерних мережах</i>	<i>29.10.2020 – 18.11.2020</i>	<i>виконано</i>
4.	<i>Охорона праці та безпека в надзвичайних ситуаціях</i>	<i>19.11.2020 – 02.12.2020</i>	<i>виконано</i>
5.	<i>Оформлення пояснювальної записки</i>	<i>03.12.2020-06.12.2020</i>	<i>виконано</i>
6.	<i>Оформлення графічного матеріалу</i>	<i>07.12.2020-11.12.2020</i>	<i>виконано</i>
7.	<i>Попередній захист дипломної роботи магістра</i>	<i>15.12.2020</i>	<i>виконано</i>
8.	<i>Захист дипломної роботи магістра</i>	<i>22.12.2020</i>	

Студент

(підпис)

Бутинець В.Ю.

(прізвище та ініціали)

Керівник проекту (роботи)

(підпис)

Луцків А.М.

(прізвище та ініціали)

АНОТАЦІЯ

Тема дипломної роботи: “Методи та засоби прогнозування трафіку комп'ютерних мереж на основі нейромережевих моделей” // Дипломна робота // Бутинець Віталій Юрійович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем та програмної інженерії, група СІМ-61 // Тернопіль, 2020 // с. – 82, рис. – 31, табл. – 12, аркушів А1 – 10, додат. – 1, бібліогр. – 28.

Ключові слова: МЕТОД, ПРОГНОЗУВАННЯ, ТРАФІК, КОМП'ЮТЕРНА МЕРЕЖА, ЧАСОВИЙ РЯД, НЕЙРОННА МЕРЕЖА.

Мета дипломної роботи магістра полягає у дослідженні методів і засобів прогнозування трафіку у комп'ютерних мережах засобами машинного навчання.

У роботі запропоновано застосування моделей ARMA та APIMA для прогнозування часових рядів при аналізі трафіку локальних комп'ютерних мереж, коли трафік передачі та приймання пакетів є однорідним і взаємодія між вузлами мережі приблизно рівномірна.

Спроектовано архітектури глибоких нейронних мереж, вхідний шар яких містить 7 нейронів, кількість проміжних шарів становить 4 рівні, вихідний шар містить один нейрон, що дало змогу практично в реальному часі забезпечити високу точність прогнозування (97%) трафіку для інтернет-сервіс провайдерів на інтервал 1 год вперед.

На основі технології контейнеризації та інструментів Docker створено контейнер з необхідною інфраструктурою для використання реалізованої моделі нейронної мережі сторонніми програмними сервісами, що дало змогу забезпечити її кросплатформність і гнучкість застосування.

ABSTRACT

The theme of the thesis: " Methods and tools of computer networks traffic forecast on neuro network models " /Master thesis / Butynets Vitalii Yuriyovych/ Ternopil Ivan Pul'uj National Technical University, Faculty of Computer Information Systems and software engineering, group CIm -61 // Ternopil, 2020// p. - 82, fig. – 31, table. – 12, Sheets A1 – 10, Add – 1, Ref. – 28.

KEYWORDS: METHOD, PREDICTION, TRAFFIC, COMPUTER NETWORK, TIME SERIES, NEURAL NETWORK.

The purpose of the master's thesis is to study the methods and means of forecasting traffic in computer networks by machine learning.

The paper proposes the use of ARMA and ARIMA models for time series prediction in the analysis of local computer network traffic, when the packet transmission and reception traffic is homogeneous and the interaction between network nodes is approximately uniform.

Architectures of deep neural networks are designed, the input layer of which contains 7 neurons, the number of intermediate layers is 4 levels, the output layer contains one neuron, which allowed to provide high real-time prediction (97%) of traffic for Internet service providers at interval 1 year ahead.

Based on the containerization technology and Docker tools, a container with the necessary infrastructure for the use of the implemented neural network model by third-party software services was created, which allowed to ensure its cross-platform and flexibility of application.

ЗМІСТ

ПЕРЕЛІК ОСНОВНИХ УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ І СКОРОЧЕНЬ	8
ВСТУП	9
РОЗДІЛ 1 АНАЛІЗ СУЧАСНИХ ДОСЛІДЖЕНЬ У ГАЛУЗІ ПРОГНОЗУВАННЯ ТРАФІКУ КОМП'ЮТЕРНИХ МЕРЕЖ	13
1.1. Аналіз особливостей формування трафіку у комп'ютерних мережах	13
1.1.1. Методи аналізу трафіку засновані на вмісті пакетів/повідомлень	15
1.1.2. Статистичні методи аналізу трафіку	18
1.2. Аналіз сучасних методів прогнозування трафіку комп'ютерних мереж	21
1.3. Аналіз функціоналів якості при прогнозуванні трафіку у комп'ютерних мережах	26
1.4. Висновки до розділу	28
РОЗДІЛ 2 ПОБУДОВА МОДЕЛІ ТА РОЗРОБКА АЛГОРИТМУ ПРОГНОЗУВАННЯ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ	29
2.1. Аналіз підходів до побудови моделі прогнозування трафіку у комп'ютерних мережах	29
2.1.1. Екстраполяція прогнозу	29
2.1.2. Регресійний аналіз	32
2.2. Нейромережеві моделі і часові ряди	33
2.2.1. Прогнозування трафіку на основі часових рядів	34
2.2.2. Прогнозування трафіку комп'ютерної мережі на основі підходу deep learning	41
2.3. Висновки до розділу	47
РОЗДІЛ 3 РЕАЛІЗАЦІЯ ЗАПРОПОНОВАНИХ МОДЕЛЕЙ ТА МЕТОДУ ПРОГНОЗУВАННЯ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ	49
3.1. Аналіз вимог, структури набору вхідних даних та проектування архітектури програмного засобу для прогнозування трафіку у комп'ютерних мережах	49
3.2. Проектування архітектури та імплементація моделі прогнозування трафіку з використанням мови програмування Python	53

3.3. Створення Docker контейнера для програмного модуля	59
3.4. Висновки до розділу	64
РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	65
4.1. Охорона праці	65
4.2. Шум, вібрація, ультразвук, електромагнітні випромінювання у виробничих приміщеннях для роботи з ВДТ та захист від них	68
ВИСНОВКИ	72
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	74
ДОДАТОК А ТЕКСТ НАУКОВИХ ПУБЛІКАЦІЙ ДИПЛОМНОЇ РОБОТИ МАГІСТРА	77

ПЕРЕЛІК ОСНОВНИХ УМОВНИХ ПОЗНАЧЕНЬ,
СИМВОЛІВ І СКОРОЧЕНЬ

ACF	AutoCorellation Function
AIC	Akaike Information Criteria
ANN	Artificial Neuro Network
BIS	Bayes Information Criteria
CNN	Convolutional Neuro Network
RNN	Recurent Neural Network
PACF	Part Auto Corellation Function
RBM	Restricted Boltzmann Machines
DBN	Deep Belief Networks
UML	Unified Modeling Language
KM	Комп'ютерні Мережі

ВСТУП

Актуальність теми. Мережі зв'язку формують базис для функціонування комп'ютерних систем загального та спеціального призначення, програмних сервісів трансферу даних, технологій обміну інформацією у військовій, банківській та інших сферах бізнесу.

Враховуючи сучасні загрози життю і здоров'ю людей, пов'язаних з поширенням пандемії COVID-19, а також тенденції до глобалізації, важливість забезпечення надійності та стабільності функціонування комп'ютерних мереж зростає в рази. Так, для провадження навчального процесу, заклади освіти інтенсивно використовують засоби дистанційного чи онлайн навчання, підприємства переходять на віддалене керування і виконання бізнес-процесів, що призводить до значного навантаження на мережі зв'язку операторів послуг, інформаційну і технічну інфраструктуру підприємства, хмарні сховища і т.д. Тому задача віддаленого моніторингу, управління та прогнозування трафіку у комп'ютерних мереж набуває все більшої актуальності. Окрім цього, алгоритми прогнозування кількості переданих та одержаних пакетів даних повинні бути адаптивними і базуватись на оптимальних інтелектуальних алгоритмах, які б давали змогу при потребі розгортати додаткову інформаційно-технічну інфраструктуру при зростанні навантаження на вузли комп'ютерних мереж, або згортати її у випадку зниження до деякого визначеного рівня.

У загальному випадку, задачі прогнозування можна розв'язувати засобами та інструментами математичної статистики, теорії імовірності, теорії оптимізації, машинного навчання та рядом інших. Варто відмітити, що дослідженню методів прогнозування у різних сферах присвячено ряд наукових та інженерних публікацій. Зокрема, серед українських вчених потрібно відмітити праці О.Г. Наконечного, Т.Б. Лобура, Г.М. Осухівської, Т. Білостоцького, А. О. Білощицького, М.З. Згуровського, Н.Д. Панкратової, П. І. Бідюка та ін, серед закордонних науковців – Н. Вінера, Е.Петерса, Дж. Мерфі, У.Ф. Шарпа, О.М. Дж. Кендала та ряду інших.

Результати, одержані науковцями, широко використовуються у сферах будівництва, прогнозування працездатності технічних систем, у системах масового обслуговування, економіці та багатьох інших сферах.

Однак, для прогнозування трафіку та навантаженості вузлів комп'ютерної мережі, потрібно провести додаткові дослідження та запропонувати оптимальне комплексне рішення, яке б давало змогу гнучко керувати трафіком та регулювати завантаженість вузлів мережі з використанням сучасних технологій, зокрема технологій машинного навчання, нейромережових моделей.

Мета і задачі дослідження. Мета дипломної роботи магістра полягає у дослідженні методів і засобів прогнозування трафіку у комп'ютерних мережах засобами машинного навчання.

Для цього в роботі розв'язуються наступні задачі:

- аналіз сучасного стану досліджень щодо прогнозування навантаженості вузлів комп'ютерної мережі і трафіку, який у ній протікає;
- дослідження факторів і критеріїв, що впливають на завантаженість вузлів комп'ютерної мережі;
- обґрунтування методів і моделей представлення та прогнозування трафіку у комп'ютерній мережі;
- розробка методу прогнозування трафіку у комп'ютерних мережах на основі часових рядів та нейромережових моделей;
- розробка прототипу програмного інструментального засобу для підтримки запропонованого методу.

Об'єкт дослідження: процес прогнозування трафіку у комп'ютерній мережі.

Предмет дослідження: моделі, методи і засоби прогнозування трафіку у комп'ютерній мережі, методи аналізу часових рядів, нейромережові моделі.

Методи дослідження: При розв'язанні задач дипломного проектування використано наступні методи:

- аналіз та узагальнення – при проведенні аналізу існуючих моделей, методів і засобів прогнозування трафіку у комп'ютерних мережах;

- теорії часових рядів, математичної статистики та машинного навчання – при побудові та верифікації моделі і методу прогнозування часових рядів;
- проектування та програмування – при створенні Docker контейнера та реалізації моделі засобами мови Python;
- експеримент – при верифікації моделей глибоких нейронних мереж із застосуванням функціоналів якості.

Наукова новизна отриманих результатів. Наукова новизна результатів дослідження полягає в наступному:

- уперше запропоновано метод прогнозування трафіку у комп'ютерних мережах на основі комплексного застосування моделей ARMA та APIMA для прогнозування часових рядів при однорідній передачі та прийманні пакетів, а також рівномірній взаємодії між вузлами мережі, що дало змогу забезпечити високу точність прогнозування трафіку у коротко і середньо-строковому інтервалах часу на рівні $\geq 95\%$ без врахування сезонної і трендової складової.
- набула подальшого розвитку модель архітектури глибокої нейронної мережі, що складається із вхідного шару з 7-ма нейронами, кількістю проміжних шарів 4 з розподілом нейронів 300-200-100-10, вихідним шаром з одним нейроном, що дало змогу практично в реальному часі забезпечити високу точність прогнозування (97%) трафіку для інтернет-сервіс провайдерів на інтервал 1 год вперед.

Практичне значення одержаних результатів. Практичне значення одержаних результатів полягає у реалізації Docker контейнера, який містить модель прогнозування трафіку у комп'ютерних мережах на коротко- і середньо-строкові інтервали часу і може виконуватись ізольовано від інших програмних додатків та незалежно від інфраструктури апаратного забезпечення.

Публікації. Результати дипломної роботи апробовані на IX міжнародній науково - технічній конференції молодих учених і студентів «Актуальні задачі сучасних технологій» (26-27 листопада 2020 р.) Тернопільського національного технічного університету імені Івана Пулюя та на VIII науково-технічній конференції Тернопільського національного технічного університету імені Івана

Пулюя «Інформаційні моделі, системи та технології» (9-10 грудня 2020 року) як тези конференцій.

1. Бутинець В.Ю., Луцків А.М. Аналіз методів прогнозування трафіку у комп'ютерних мережах. Матеріали ІХ міжнародної науково - технічної конференції молодих учених і студентів «Актуальні задачі сучасних технологій» (26-27 листопада 2020 р.) Тернопільського національного технічного університету імені Івана Пулюя. Тернопіль: ТНТУ. 2020. С. 40.

2. Бутинець В.Ю., Луцків А.М. Методи аналізу трафіку у комп'ютерних мережах. Матеріали VIII науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі, системи та технології» (9-10 грудня 2020 року). Тернопіль: ТНТУ. 2019. С. 201.

Структура роботи. Дипломна робота містить розрахунково-пояснювальну записку та графічний матеріал. До складу записки входить вступу, 4 розділи, загальні висновки, список використаних джерел і додатки. Обсяг роботи: розрахунково-пояснювальна записка – 82 арк. формату А4, графічна частина – 10 аркушів формату А1.

РОЗДІЛ 1

АНАЛІЗ СУЧАСНИХ ДОСЛІДЖЕНЬ У ГАЛУЗІ ПРОГНОЗУВАННЯ ТРАФІКУ КОМП'ЮТЕРНИХ МЕРЕЖ

1.1. Аналіз особливостей формування трафіку у комп'ютерних мережах

На сьогоднішній день проектування, впровадження та експлуатація комп'ютерних мереж є актуальною задачею для багатьох сфер, що обумовлено необхідністю автоматизації бізнес процесів та віддаленого виконання посадових обов'язків в умовах карантину. Комп'ютерні мережі дають змогу забезпечити єдиний інформаційний простір та налаштувати права доступу до визначених ресурсів. При цьому важливу роль відіграє правильна організація доступу до мережі Інтернет.

На базі комп'ютерної мережі можна організувати ефективний обмін даними, забезпечити продуктивність використання апаратних ресурсів для розв'язання складних математичних задач. Для освітнього простору комп'ютерні мережі виконують функцію доступу до навчальних матеріалів, проведення відеоконференцій та тестування знань. На виробництві КМ забезпечують віддалений доступ до керування периферійними пристроями, вузлами агрегатів, верстатами та ін.

Об'єднання комп'ютерів у мережі надає наступні переваги [1]:

- ефективне використання дискового простору, що полягає у спільному використанні необхідних ресурсів багатьма користувачами системи;
- гнучкість маршрутизації при збоях на окремих вузлах комп'ютерної мережі;
- ефективність резервного зберігання даних та резервування каналів зв'язку;
- здатність до побудови на існуючому апаратному забезпеченні віртуальних машин та мереж;

– можливість авторизованого доступу до обладнання та ресурсів комп'ютерної мережі.

У свою чергу різновид комп'ютерних мереж – локальні комп'ютерні мережі за функціональним призначенням та територіальним розташуванням характеризуються наступними критеріями [3]:

- територія розгортання мережі в межах однієї організації, а відстань між вузлами мережі – в межах до кількох кілометрів;
- забезпечення багатокористувацького доступу до ресурсів, що розміщені на вузлах мережі;
- управління правами доступу, які забезпечує адміністратор мережі;
- забезпечення стабільності функціонування сервісів, надання яких передбачено у комп'ютерній мережі;
- фізичне з'єднання вузлів на деякій локальній території.

Таким чином, комп'ютерні мережі дають змогу знизити витрати на збір, опрацювання та передачу інформації. Ще однією важливою перевагою КМ є те, що зменшується відсоток непродуктивних витрат робочого часу співробітниками підприємства. Такі переваги забезпечують комп'ютерам, які об'єднані в локальну мережу, гнучкість роботи та оптимальний розподіл ресурсів, програм та даних.

Однією із найбільш важливих характеристик комп'ютерної мережі є її топологія. Ця характеристика визначає спосіб організації зв'язків між вузлами мережі та їхню структуру. В загальному випадку розглядають фізичну та логічну топології мережі, які відрізняються між собою способом представлення взаємодії між компонентами.

Основне завдання фізичної топології надати достовірну інформацію про типи і способи фізичного з'єднання компонентів комп'ютерної мережі. На противагу фізичній топології, логічна топологія представляє комп'ютерну мережу на рівні логічної взаємодії її компонентів.

Комплексно фізична та логічна топології утворюють базис, який дає змогу більш повно і широко представляти комп'ютерну мережу. Маючи фізичну та

логічну топології мережі можна проводити її аналіз щодо пропускнуої здатності, навантаження на вузли мережі та прогнозувати трафік – кількість пакетів, які циркулюють у мережі.

В загальному випадку, IP-трафік у комп'ютерних мережах можна поділити на групи, як показано на рис. 1.1.

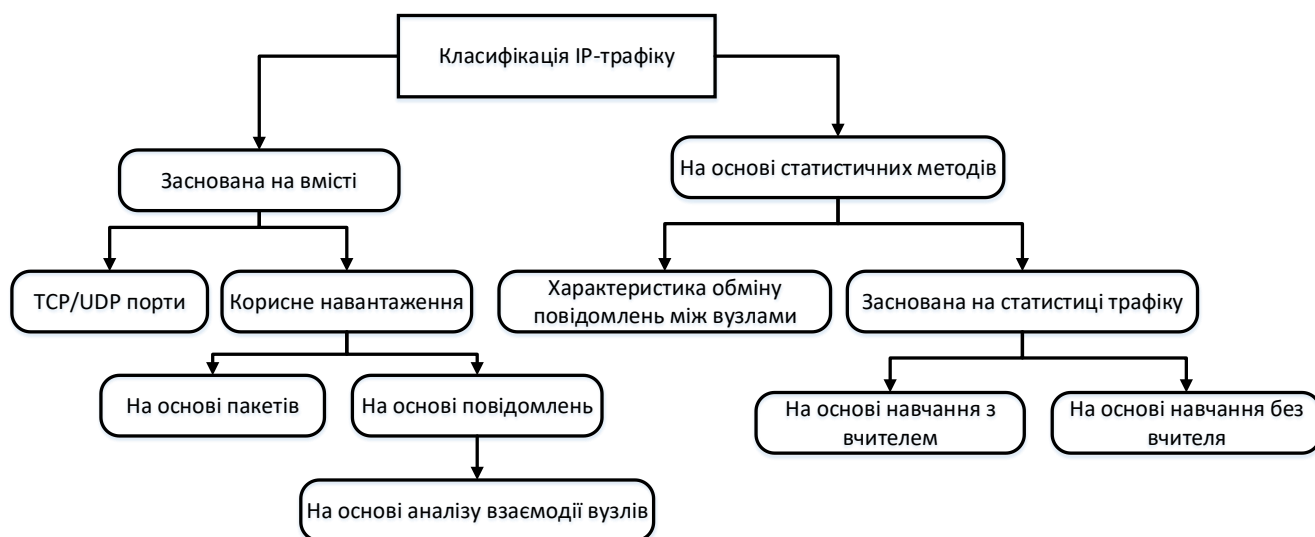


Рис. 1.1. Методи класифікації IP-трафіку

Трафік на практиці поділяють на класи в залежності від типу даних і протоколів передачі, які використовуються. Проведемо аналіз методів класифікації IP трафіку, які засновані на аналізі вмісту пакетів/повідомлень і статистичних методів.

1.1.1. Методи аналізу трафіку засновані на вмісті пакетів/повідомлень

Більшість програмних продуктів, які функціонують у комп'ютерній мережі передають дані через вже відомі порти апаратного забезпечення. У такому випадку, суть задачі класифікації пакетів полягає у знаходженні TCP SYN-пакету для встановлення серверної частини клієнт-серверного з'єднання на основі TCP. Після цього, необхідно одержати результат щодо доступності цільового номера порту для

конкретного додатку, який надіслав пакет. По такому ж принципу працює класифікація пакетів на основі UDP, однак з'єднання при цьому не встановлюється.

Найбільш вагомою перевагою даного методу є простота реалізації та швидкість виконання операцій щодо класифікації. Однак даний метод має і ряд недоліків. До них належать:

- відсутність у деяких програмних продуктів власних портів, які зареєстровані в IANA (функція управління IP адресним простором);
- здатність програмних додатків використовувати відмінні від визначених в операційній системі портів для виконання певних функцій, наприклад, використання відмінного від порта «80» в Unix-подібних системах для HTTP-сервера;
- виникнення помилок при шифруванні на IP рівні, що може спровокувати плутанину TCP і UDP заголовків.

Інший підхід базується на тому, що для зменшення залежності від портів і одержання достовірних даних про використовуваний протокол, застосовуються методи відновлення стану сеансу та аналізуються дані щодо вмісту кожного окремого пакету.

Класифікація на основі корисного навантаження для peer-to-peer трафіку передбачає аналіз його сигнатур на прикладному рівні. Застосування такої класифікації пакетів дає змогу зменшити кількість помилок першого і другого роду до 5%.

Класифікацію, базовану на корисному навантаженні, можна організувати шляхом застосування наступних методів перевірки та опрацювання:

- PBNS (Packet Based No State);
- PBFS (Packet Based Per Flow State);
- MBFS (Message Based Per Flow State);
- MBPS (Message Based Per Protocol State).

Окрім цього, існує чотири шляхи перевірки при класифікації корисного навантаження. Перша метрика, яка може бути використана при перевірці – знаходження деякої сигнатури корисного навантаження на прикладному рівні. Для

прикладу, структура пакету HTTP починається після URL-адреси і даних про версію протоколу, а до структури Edonkey-пакетів входять поля, що містять інформацію про корисне навантаження.

Наступна перевірка пакетів полягає у синтаксичному аналізі повідомлень, що є різновидом сигнатурної перевірки та орієнтована на верифікацію правильності переданих даних в контексті синтаксичної структури. Для прикладу, HTTP заголовки повинні бути наявними у відповідному корисному навантаженні. Така перевірка передбачає процес декодування усіх полів повідомлення.

Третій вид контролю тісно пов'язаний з протоколом відповідності, зокрема, передбачається, що на деякий запит (request) клієнта завжди буде відповідь від сервера (response). Така форма перевірки є більш точнішою тому, що здійснює верифікацію поведінки протоколу з його специфікацією.

Ще один вид контролю передбачає семантичну перевірку даних. Суть такої верифікації полягає у визначенні типу об'єкту, що передається, зокрема чи є дані, наприклад, зображенням, відео чи іншим видом інформації. Семантичний аналіз на сьогодні є найбільш не формалізованим методом, однак дає змогу визначати ефект «тунелювання».

Метод PBNS здійснює перевірку сигнатури за визначеними полями, які притаманні будь-якому пакету. За рахунок того, що PBNS досить простий з позиції обчислень, тобто опрацьовуються лише деякі поля у пакеті, то він не вимагає зберігання станів.

В основі методу PBFS лежить аналіз таблиці сеансів. При цьому кожен сеанс містить інформацію про його ідентифікатор та ідентифікатор протоколу прикладного рівня. Ідентифікатор сеансу формується з IP адрес відправника/одержувача, ідентифікатора транспортного протоколу, портів відправника/одержувача. Таблиця сеансів по об'єму становить кілька десятків байтів.

Метод MBFS базується на повідомленнях і передбачає використання модуля нормалізації TCP/IP пакетів. Суть роботи такого методу подібний до перевірки корисного навантаження на основі PBFS, але відмінність полягає в тому, що MBFS

працює на рівні повідомлень, а не пакетів. Відповідно вимагається більше ресурсів пам'яті, у зв'язку з наявністю додаткової інформації про сеанси, їхні стани та буферизацію при функціонуванні TCP/IP нормалізаторів. Параметри MBFS залежать від типу трафіку і фрагментації пакетів TCP сеансів.

MBPS методи здатні забезпечити високу точність інтерпретації повідомлень, які передаються і приймаються програмними додатками. Модуль опрацювання MBPS здатний аналізувати не тільки семантику повідомлення, але й етапи передачі повідомлень. Характерною особливістю застосування цих методів є значне зростання ресурсів пам'яті, що пов'язано з необхідністю аналізу трафіку на транспортному та прикладному рівнях. Ефективність MBPS, у порівнянні з іншими методами, є найбільш високою, оскільки усі дані прикладного рівня повинні бути опрацьованими і таким чином забезпечена точність класифікації повідомлень.

1.1.2. Статистичні методи аналізу трафіку

Статистичні методи дають змогу аналізувати трафік у комп'ютерних мережах за двома підходами:

- на основі алгоритмів аналізу поведінки трафіку;
- статистичні методи аналізу трафіку на мережевому і транспортному рівнях.

Основна ціль алгоритмів аналізу поведінки трафіку ґрунтується на визначенні додатків, які генерують найбільшу частину трафіку у комп'ютерній мережі. У результаті такого аналізу можна виявити, яким чином вузли комунікують між собою і яке програмне забезпечення функціонує на них. Основним предметом дослідження, в даному випадку, виступають статистичні характеристики трафіку, які дають змогу ідентифікувати додаток, що генерує трафік. При цьому припускають, що кожен клас програмного забезпечення на мережевому рівні володіє характеристиками, які є унікальними для різних класів ПЗ та існує можливість ідентифікації додатку у межах класу.

В загальному випадку, статистичні методи аналізу трафіку побудовані на принципах класичного машинного навчання, які використовують методи класифікації з вчителем та без нього (методи кластерного аналізу).

Для розв'язання задачі класифікації при аналізі IP-трафіку, об'єкти, які досліджуються, набувають деяких інших семантичних відтінків. Термінологічний словник, при цьому передбачає використання наступних понять:

- під простим потоком, у даному випадку, розуміють сукупність послідовних однонаправлених пакетів, які характеризуються п'ятьма ознаками: IP-адреса відправника, IP-адреса одержувача, номер порту відправника, номер порту одержувача, ідентифікатор протоколу;
- двонаправлений потік – представляється у вигляді пари простих потоків різної направленості між тими самими відправниками та одержувачами;
- повний потік – двонаправлений потік, який існує протягом сеансу взаємодії між відправником та одержувачем.

Загальний процес бінарної класифікації трафіку (трафік від відомого ПЗ або іншого трафіку) наведено на рис. 1.2, який передбачає наявність вхідної множини даних про трафік у комп'ютерній мережі, процедури розбиття на навчальну і тестову вибірку, побудову моделі та алгоритмів класифікації.

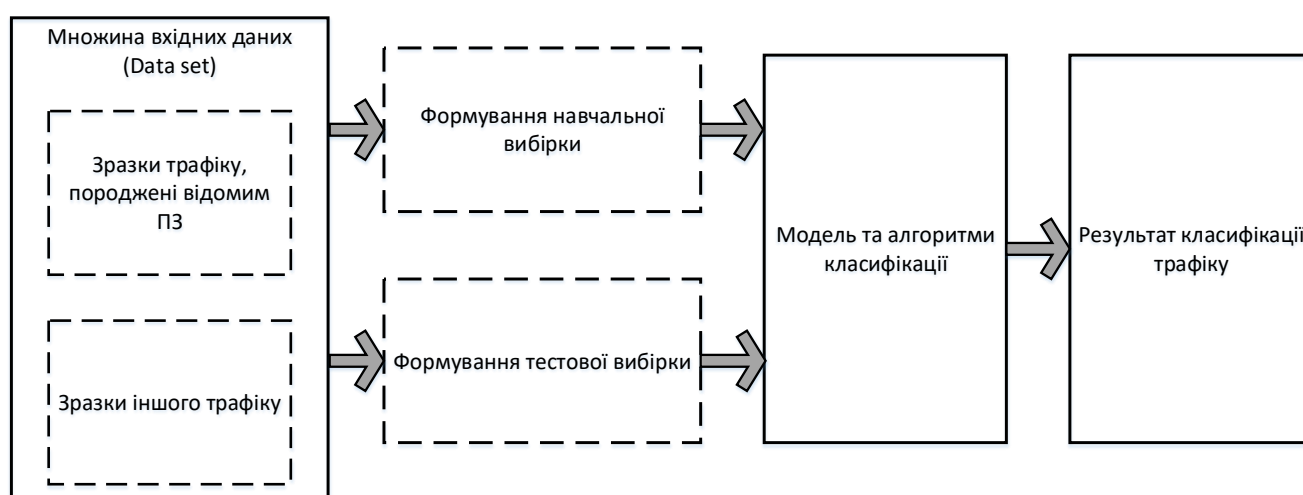


Рис. 1.2. Процедура бінарної класифікації з використанням алгоритмів навчання з вчителем

На рис. 1.3. наведено більш детальний процес навчання і тестування, які притаманні класифікації трафіку у комп'ютерній мережі. При використанні алгоритму навчання з учителем обов'язковою є наявність множини вхідних даних з наперед визначеними класами двох типів IP трафіку.

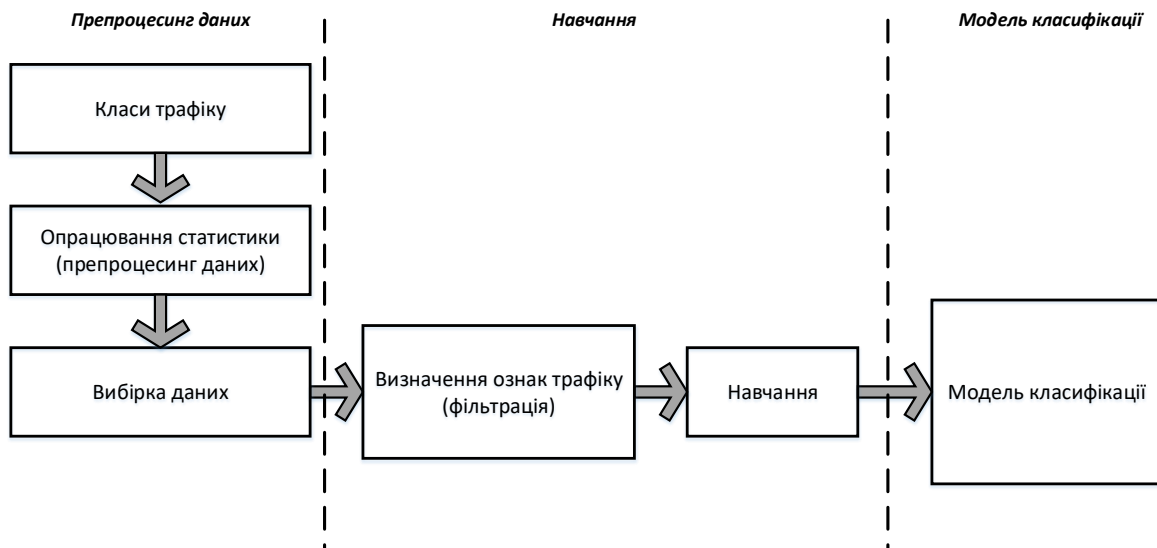


Рис. 1.3. Класифікатор з учителем

На рис. 1.4 наведено процедуру алгоритму класифікації, що описує послідовність кроків при навчанні з учителем.

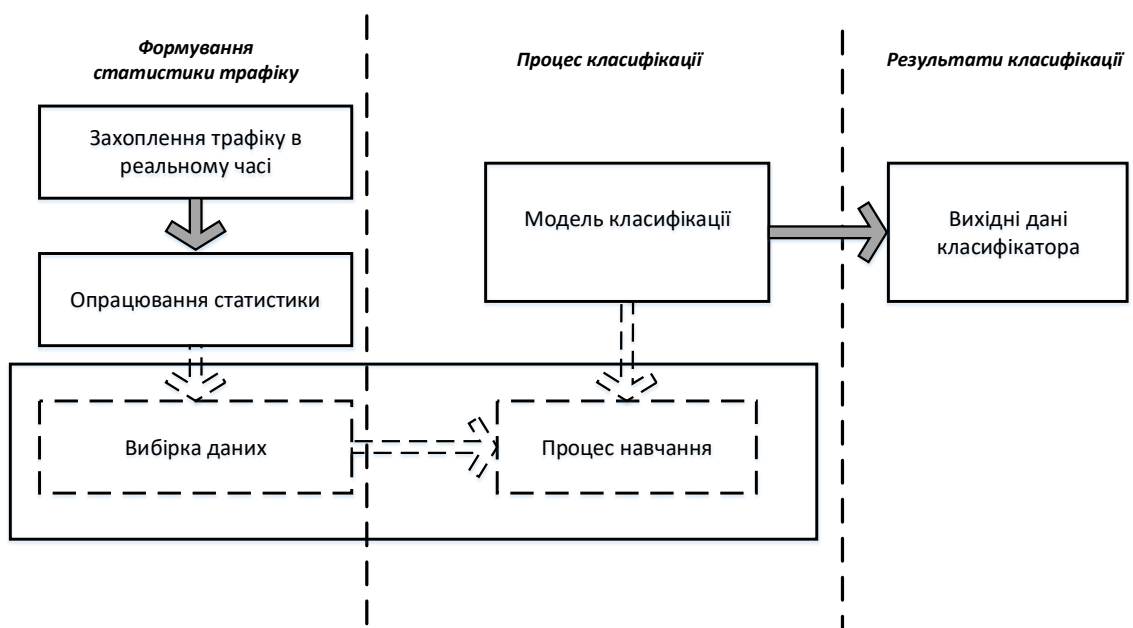


Рис. 1.4. Процедура класифікації трафіку в реальному часі

Як видно з рис. 1.4, спочатку формується статистика трафіку на основі функціонування екземплярів програмного забезпечення, яке становить цільову змінну, та представники інших програмних продуктів, які при передачі даних можуть використовувати, наприклад, HTTP. Опрацювання статистики трафіку передбачає визначення ознак трафіку.

Трафік, який формується у реальному часі, є базою для визначення ознак потоку, який подається на алгоритм класифікації. Тому, використання підходу на основі машинного навчання, що передбачає модель навчання з учителем є ефективним засобом не тільки при ідентифікації типу трафіку, але й при його прогнозуванні.

1.2. Аналіз сучасних методів прогнозування трафіку комп'ютерних мереж

Сучасний розвиток інформаційних технологій, комп'ютерної інженерії, інженерії програмного забезпечення, системного аналізу дає змогу розв'язувати задачі прогнозування для різних сфер шляхами, відмінними від тих, які використовувались до цього часу. До традиційних моделей належать регресійні моделі та моделі на основі часових рядів. До не традиційних – моделі і методи експертного оцінювання та нейромережевих технологій.

Такі моделі відрізняються від традиційних тим, що не вимагають побудови моделі об'єкта у класичному розумінні та не втрачають стійкість і функціональність при неповній вхідній інформації.

Методом прогнозування, згідно [1], – це спосіб вивчення об'єкта, який спрямований на створення прогнозу. Для того, щоб обрати обґрунтовано оптимальний метод прогнозування трафіку комп'ютерної мережі проведемо аналіз існуючих методів прогнозування. В загальному випадку, методи прогнозування можна поділити на класи, як показано на рис. 1.5.

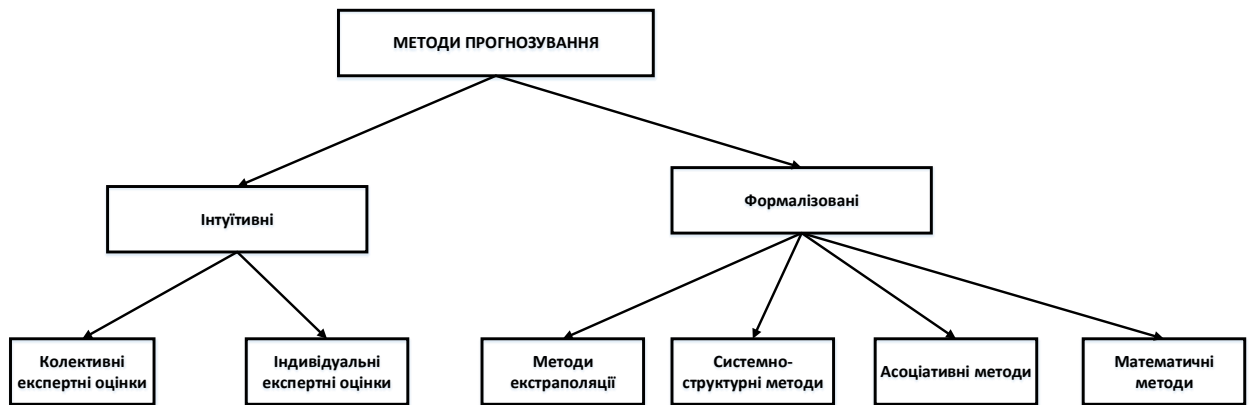


Рис. 1.5. Методи прогнозування

З рис. 1.5 видно, що методи прогнозування можна поділити на дві основні групи: інтуїтивні та формалізовані. Інтуїтивні методи (експертні методи) застосовуються у випадках, коли об'єкт прогнозування дуже простий, або навпаки занадто складний і аналітично спрогнозувати вплив різних факторів на об'єкт практично неможливо. Основу функціонування таких методів забезпечують експертні технології, які враховують думку одного або кількох експертів в конкретній галузі дослідження.

Результати досліджень щодо прогнозування трафіку у комп'ютерних мережах показують, що для прогнозування кількості пакетів переданих і одержаних даних не існує стандартного, єдиного методу: будь-яке підприємство характеризується своїми особливостями організації бізнес-процесів, які формують деякий унікальний процес. Однак, для більшості підприємств, закладів освіти, вищих навчальних закладів, можна визначити фактори, які впливають на навантаження, тобто кількість трафіку, у комп'ютерній мережі. Чим більше інформації існує про навантаження вузлів у комп'ютерній мережі, тим точніший можна побудувати прогноз.

Процеси формування трафіку у комп'ютерних мережах мають функціональні, циклічні і випадкові тенденції. Найпростіше піддаються прогнозуванню циклічні залежності (зазвичай добові, тижневі і річні). Циклічні тенденції, за попередніми оцінками, становлять близько 70 - 80% всіх змін при формуванні навантаження на

вузли комп'ютерної мережі. Наприклад, найбільш істотними циклічними факторами вважають час і день тижня.

Наступна важлива група факторів, які відносяться до основних факторів впливу на трафік у комп'ютерній мережі і досліджується при прогнозуванні навантаження на вузли мережі, є закономірності функціонального характеру. Їх частка становить близько 10 - 15% від загального обсягу відхилень. До цієї групи факторів можуть бути включені відхилення, які пояснюються відомими і передбачуваними факторами, що є характерними для конкретного об'єкту господарювання: кількість одночасно виконуваних проектів, кількість одночасно триваючих занять, пропускна здатність вузлів мережі, здатність до автоматичного розгортання віртуальних сервісів і т.д. Аналіз даних допомагає виявити ці фактори і визначити їхній ваговий коефіцієнт у процесі формування та прогнозування трафіку комп'ютерної мережі.

Розв'язавши задачі впливу функціональних та циклічних факторів, на практиці може виявитися що існує проблема недостатності або недостовірності вхідних даних: недоступність інформації, помилкові дані або затримка інформації про обсяги навантаження на вузли мережі. Як наслідок, все це призводить до отримання недостовірного прогнозу щодо трафіку у комп'ютерній мережі.

Третя група факторів, які впливають на прогнозування трафіку - випадкові тенденції чи ситуації. Їх частка в загальному процесі як правило невелика, але амплітуда відхилень може бути значною. До таких факторів можна віднести аварійне раптове відключення електроенергії, вихід з ладу вузлів комп'ютерної мережі, недоцільність формування прогнозів трафіку у вихідні чи святкові дні. Тому при прогнозуванні необхідно враховувати випадкові фактори, які вносять імовірнісні характеристики відповідних явищ.

У зв'язку з відсутністю універсальної методики і універсального програмно-апаратного засобу для вирішення задачі прогнозування трафіку у комп'ютерних мережах, кожен суб'єкт господарювання розв'язує її власними методами з урахуванням особливості бізнес-процесів, або не враховує взагалі. Важливим етапом прогнозування є оцінка адекватності моделі. При розробці моделі

прогнозування вхідні дані поділяються на дві частини, одна з яких охоплює більш ранні дані, а інша – історичні, більш пізні. За допомогою даних першої групи оцінюються параметри моделі прогнозу, а дані другої групи розглядаються як фактичні дані прогнозованого показника. Після розробки моделі прогнозування отриманий прогноз порівнюють з даними другої групи, для оцінки його розробленої моделі, тобто відповідності фактичних статистичних даних. Крім того, коли будують кілька моделей прогнозування, не завжди очевидно, яка з них краща і за якими критеріями. Для оцінки точності моделей застосовують різні критерії [2]. При роботі з регресійними моделями і, зокрема, для їх перевірки виконують наступні етапи [3]:

- перший етап - перевірка статистичної значущості отриманих коефіцієнтів рівняння регресії;
- другий етап - перевірка загальної якості побудованого рівняння регресії;
- третій етап - перевірка властивостей даних при оцінюванні рівняння.

Для визначення адекватності одержаного рівняння регресії досліджуваного процесу чи об'єкту, можливий один з наступних варіантів:

1. Побудована модель на основі F-критерію Фішера вважається адекватною і всі коефіцієнти регресії значущі. Така модель є ефективною, і її можна використовувати для прийняття управлінських рішень і створення прогнозів.

2. Побудована модель за F-критерієм Фішера в цілому адекватна, але частина коефіцієнтів є незначущими. Модель придатна для прийняття деяких рішень, але для прогнозування така модель не може бути використана.

3. Побудована модель за F-критерієм Фішера в цілому адекватна, але всі коефіцієнти регресії є незначущими. Така модель визнається повністю неадекватною і її відправляють на додаткові дослідження.

Для того, щоб оцінити значущість рівняння регресії в цілому, використовують критерій Фішера, критерію Фішера передуює дисперсійний аналіз.

Дисперсійний аналіз в математичній статистиці може використовуватися як самостійний інструмент для проведення статистичного аналізу. В економетриці дисперсійний аналіз частіше застосовують як додатковий засіб для дослідження

якості побудованої регресійної моделі. Принцип основної ідеї дисперсійного аналізу: загальна сума квадратів відхилень змінної (y) від середнього значення розкладається на дві складові: «зрозумілу» і «незрозумілу» [10]

$$\sum(y - \bar{y})^2 = \sum(\widehat{y}_x - \bar{y})^2 + \sum(y - \widehat{y}_x)^2 \quad (1.1)$$

де \widehat{y} – середнє значення ряду. Відношення «зрозумілої» частини дисперсії (змінної y) до загальної дисперсії називають коефіцієнтом детермінації і використовується для оцінки якості отриманого рівняння регресії. Співвідношення між «зрозумілою» і «незрозумілою» частинами загальної дисперсії можна обчислити за альтернативними формулами:

$$R^2 = \frac{\sum(\widehat{y} - \bar{y})^2}{\sum(y_i - \bar{y})^2} = 1 - \frac{\sum(y_i - \widehat{y}_x)^2}{\sum(y_i - \bar{y})^2} \quad (1.2)$$

Область визначення коефіцієнту детермінації R^2 належить інтервалу від нуля до одиниці $0 \leq R^2 \leq 1$. За допомогою коефіцієнта детермінації R^2 визначають, яку частину дисперсії результату (y) змогло пояснити рівняння регресії. Чим більше R^2 , тим краще рівняння описує вхідні дані і тим більша частина дисперсії результативної ознаки (y). За відсутності у результаті дослідження залежності між (y) і (x) коефіцієнт детермінації R^2 буде близький до нуля. Таким чином, коефіцієнт детермінації R^2 може застосовуватися для оцінки якості отриманого рівняння регресії. Використання коефіцієнта детермінації R^2 для оцінки якості моделі, має недолік, якщо необхідне включення в модель нового фактора (навіть малознащущого), то це автоматично збільшує величину R^2 . На практиці існує ряд ситуацій, в яких наведена формула для розрахунку помилок не дає достовірної інформації про властивості побудованих моделей. Тому для отримання більш об'єктивної оцінки якості моделі варто розраховувати кілька коефіцієнтів.

1.3. Аналіз функціоналів якості при прогнозуванні трафіку у комп'ютерних мережах

Аналіз оцінок прогнозування показав, що найбільш широко застосовуються наступні п'ять функціоналів якості: MAPE, MAD, MSE, SSE, MPE, MSEN [11]. Проаналізуємо їх особливості: MAPE (the mean absolute percentage error), середній абсолютний відсоток помилки (середня відносна помилка прогнозу)

$$MAPE = \frac{1}{n} \sum_{t=1}^n \varepsilon_t \quad (1.3)$$

де n - кількість спостережень. Дана оцінка застосовується для часових рядів, фактичні значення яких значно більші 1. Наприклад, оцінки помилки прогнозування трафіку у комп'ютерних мережах майже у всіх дослідженнях наводяться як значення MAPE.

$$e_t = y_t - y_t^{sim^2} \quad (1.4)$$

де y_t – фактичне значення показника на момент часу t ;

$y_t^{sim^2}$ – значення показника, одержаного за допомогою моделі, на момент часу t .

MAD (mean absolute deviation) – середня абсолютна похибка, яка розраховується за формулою

$$MAD = \frac{1}{n} \sum_{t=1}^n |Y_t - \widehat{Y}_t| \quad (1.5)$$

Середня абсолютна похибка показує, на скільки в середньому відхиляється в більшу або меншу сторону прогноз. MAD дозволяє визначити помилку в конкретних одиницях.

MSE (Mean square error) – середньоквадратична помилка моделі регресії. Хороша якість наближення даних параметричної моделі показує близькість MSE до нуля

$$MSE = \sqrt{\frac{\sum_{i=1}^n e_i^2}{n-h}} \quad (1.6)$$

де h – це кількість параметрів, що входять в модель регресії.

SSE (Sum of Square Error) – квадратична помилка.

$$SE = \sum_{t=1}^n (Y_t - \hat{Y}_t)^2 \quad (1.7)$$

У даному випадку близькість SSE до нуля говорить про хорошу якість наближення даних параметричної моделі.

MPE (mean percentage error), середній відсоток помилки розраховується за формулою

$$MPE = \frac{1}{n} \sum_{t=1}^n \frac{e_t}{y_t} \times 100\% \quad (1.8)$$

Наприклад, MPE характеризує відносний ступінь зміщення прогнозу. Високе значення цього критерію одержують, коли пов'язані із заниженням фактичного прогнозованого значення, втрати при прогнозуванні, врівноважуються завищенням. Ефективний прогноз повинен бути незміщеним, і його значення MPE повинно наближатися до нуля.

Середня відсоткова помилка показника не повинна перевищувати 5%. Отримані значення якості інформаційної придатності побудованої моделі прогнозування показують, на скільки достовірна модель і як вона описує в подальшому значення досліджуваного явища чи об'єкта, в даному випадку трафіку або навантаження вузлів у комп'ютерній мережі. Чим точніше побудована модель описує історичні значення, тим вище ймовірність того, що вона буде передбачати

майбутні значення з високою точністю. Для визначення надійності моделі прогнозування необхідно порівнювати фактичні і прогнозовані значення.

1.4. Висновки до розділу

1. Проведено аналіз принципів організації та особливостей формування трафіку у комп'ютерних мережах, що дало змогу визначити фактори і можливі методи класифікації потоків даних, які циркулюють у мережі. Встановлено, що найбільш ефективним способом визначення типу трафіку є статистичні моделі класифікації з учителем, які в подальшому можна використати при прогнозуванні трафіку.

2. Проаналізовано сучасні підходи, моделі і методи прогнозування трафіку у комп'ютерних мережах і встановлено, що в залежності від терміну прогнозування та кількості наявних історичних даних, найбільш ефективними є регресійні моделі, моделі і методи аналізу часових рядів, нейромереві підходи.

3. Проведено аналіз функціоналів якості при прогнозуванні трафіку у комп'ютерних мережах і запропоновано використовувати не один, а декілька показників якості, що дає змогу оцінити точність алгоритму прогнозування, його робастність (стійкість результатів прогнозування) та інші.

РОЗДІЛ 2

ПОБУДОВА МОДЕЛІ ТА РОЗРОБКА АЛГОРИТМУ ПРОГНОЗУВАННЯ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ

2.1. Аналіз підходів до побудови моделі прогнозування трафіку у комп'ютерних мережах

Згідно [4], часовий ряд представляє собою послідовність будь-яких значень деякої величини у різні моменти часу.

Випадковим рядом можна вважати кількість переданих чи/та одержаних даних у комп'ютерній мережі або кількість користувачів, які створюють навантаження на вузли мережі деякого об'єкта (поштовий сервер, сервер дистанційного навчання, файловий сервер, маршрутизатор і т.п.). Часовий ряд як правило включає в себе вибірку значень навантаження вузол комп'ютерної мережі у різні моменти часу або обсяг переданих пакетів даних за певний інтервал часу.

Формалізованими методами прогнозування часових рядів є:

- екстраполяція прогнозу;
- регресійний аналіз (прогнозування з використання штучних нейронних мереж; прогнозування на базі АРІМА моделей);
- адаптивні методи прогнозування;
- прогнозування з використанням гібридних систем.

Для прогнозування трафіку у комп'ютерних мережах проведемо аналіз вище зазначених методів і визначимо можливість їх застосування.

2.1.1. Екстраполяція прогнозу

Один з найпоширеніших і простих методів є екстраполяція при проведенні прогнозування. Для побудови прогнозів за допомогою екстраполяції збирають статистично важливі тенденції зміни тих чи інших кількісних характеристик досліджуваного об'єкта.

Для використання екстраполяції потрібно мати інформацію про історію розвитку об'єкта за період, в 2-3 рази більший періоду прогнозування.

Цей метод можна, використовувати для короткострокового прогнозування. Також до методів екстраполяції відносяться такі методи:

- метод найменших квадратів та його різновиди;
- метод ковзного середнього;
- метод експоненційного згладжування.

Метод найменших квадратів передбачає застосування мінімізації суми квадратичних відхилень між фактичними і розрахунковими значеннями.

Метод ковзного середнього застосовується, коли в рядах спостерігаються різкі коливання параметрів за періодами. У таких рядах, як правило, слабкий зв'язок параметрів з часом і неможливо побачити та дослідити тенденції до зміни параметрів. Зазвичай, для простоти розв'язку такої задачі, використовується метод згладжування або вирівнювання динамічного ряду. Існують багато способів для забезпечення згладжування, суть яких зводиться до заміни фактичних рівнів динамічного ряду розрахунковими, тобто зниженням коливанням у порівнянні з вхідними даними. Реалізацію згладжування можна забезпечити шляхом застосування ковзного середнього, що дає змогу згладити періодичні випадкові коливання і таким чином визначити наявний тренд досліджуваного динамічного ряду. За допомогою методу ковзного середнього можна виявити та ліквідувати випадкові коливання ряду завдяки заміні значень деякого діапазону значеннями середнього арифметичного.

Метод експоненціального згладжування був запропонований Р. Брауном. На основі цього методу можна одержати оцінку параметрів тренду, що відображає не середній рівень процесу, а тенденцію, що спостерігається до моменту крайнього спостереження.

Найбільш часто метод експоненціального згладжування застосовується, коли досліджувані значення параметрів об'єкту мають помірний зв'язок у часі і можуть давати зрозумілі прогнози, а також забезпечують врахування показників, які одержують у період спостереження. Згладжування відбувається за допомогою

зваженого середнього ковзного динамічного ряду, коли ваги параметрів мають експоненційний розподіл.

Метод експоненціального згладжування не тільки використовує виявлені попередні залежності, але й може адаптовуватись під умови, які змінюються в часі. Даний метод є ефективним засобом при формуванні середньо строкового прогнозу.

Часовий ряд при застосуванні екстраполяції можна представити у вигляді

$$y_t = f(t) + \varepsilon_t \quad (2.1)$$

де $f(t)$ – деяка не випадкова функція від часу;

ε_t – випадкова величина з нульовою середньою дисперсією $D(\varepsilon_t)$.

По іншому, випадковий ряд можна записати у вигляді

$$y_t = f(t) + S + C + \varepsilon_t \quad (2.2)$$

де $f(t)$ – детермінована не випадкова функція або по іншому тренд;

S – сезонна складова;

C – циклічна складова;

ε_t – стохастична компонента процесу.

Детермінована компонента $f(t)$ визначає існуючу динаміку розвитку процесу в цілому, а стохастична ε_t – показує випадкові коливання, тобто шуми у процесі. Звідси випливає, що прогнозування часового ряду зводиться до визначення детермінованої компоненти, сезонної, циклічної і стохастичної складових.

До переваг наведеного методу можна віднести простоту реалізації моделі прогнозування, а також можливість його застосування з мінімальним набором вхідних параметрів.

До недоліків застосування методу екстраполяції варто віднести наступні:

- неможливість врахувати багатьох параметрів, які впливають на трафік комп'ютерної мережі та навантаження на окремі вузли;
- чітка статична фіксація тренду, яка не може врахувати короткострокові тенденції щодо формування трафіку у комп'ютерній мережі, що пов'язані із зростанням кількості користувачів системи, обсягів переданих та одержаних даних.

Метод екстраполяції може бути ефективно застосований при прогнозуванні трафіку у комп'ютерних мережах із стабільними та регулярними змінами кількості пакетів даних.

2.1.2. Регресійний аналіз

Регресійний аналіз входить у розділ математичної статистики, що об'єднує методи дослідження регресійної залежності між величинами за статистичними даними. При використанні підходу регресійного аналізу визначається загальний вид регресійного рівняння (рівняння регресії), проводиться розрахунок невідомих параметрів, які входять у рівняння регресії та виконується перевірка статистичних гіпотез щодо регресії.

Задача регресійного аналізу розв'язується за декілька кроків:

- опрацювання вхідних даних;
- визначення виду рівняння регресії і розрахунок її коефіцієнтів;
- оцінка адекватності побудованої моделі за результатами спостереження.

Перший крок передбачає стандартизацію матриці вхідних даних, розрахунок коефіцієнтів кореляції, перевірку їх значущості і при потребі видалення незначимих властивостей об'єкту.

У результаті виконання першого кроку одержують стандартизовану матрицю спостережень і кореляційну матрицю. Стандартизовану матрицю, з точки зору геометрії, можна інтерпретувати наступним чином: в n -вимірному просторі осі відповідають вихідним даним і показнику. Матриця представляє сукупність n векторів у просторі параметрів, а кожен рядок цієї таблиці – вектор у n вимірному

просторі. Осі n -вимірного простору відповідають результатам окремих спостережень. Вектори у досліджуваному просторі спостережень відповідають стовпцям матриці. Всі вектори мають однакову довжину, яка дорівнює \sqrt{n} .

Кут між двома векторами визначає взаємозв'язок відповідних величин. Чим менший кут між векторами, тим більший зв'язок між властивостями об'єкту, тобто більший коефіцієнт кореляції.

Одним з поширених методів одержання прогнозів щодо навантаження вузлів комп'ютерної мережі є метод найменших квадратів багатокритеріальної моделі регресії. У випадку лінійної регресії, її рівняння можна записати у вигляді [6]

$$y_j = \sum_{i=1}^n a_i x_{ij} + \varepsilon_j \quad (2.3)$$

де a_i – коефіцієнти регресії;

y_j, x_{ij} – значення j – ої функції (залежної змінної) та i – ої незалежної змінної (регресора);

ε_j – випадкова помилка;

n – кількість незалежних змінних у моделі.

У більшості випадків вважають, що a_i – вільний член, а $x_{0j} = 1$. Для побудови моделі регресії необхідно знайти її коефіцієнти на основі даних, що описують об'єкт прогнозування і фактори, що на нього впливають. На практиці, більшість моделей прогнозування базуються саме на цьому методі. У результаті прогнозування трафіку комп'ютерних мереж на основі таких моделей припускають сталість значень коефіцієнтів регресії у часі.

2.2. Нейромеревеві моделі і часові ряди

До сучасних технологій прогнозування трафіку у комп'ютерних мережах належать підходи, що базуються на застосуванні нейромерев. Вони є апробованими у ряді областей: статистика, нейрофізіологія, комп'ютерні науки,

математика, фізика. Технології нейромереж застосовують при розпізнаванні образів, моделюванні об'єктів предметних областей, при аналізі часових рядів, опрацюванні сигналів та інших.

2.2.1. Прогнозування трафіку на основі часових рядів

Моделі серії ARIMA можна обрати як базис для прогнозування трафіку у комп'ютерних мережах. В основі цього методу лежить дослідження функцій автокореляції та часткової автокореляції (ACF та PACF відповідно) [5]. Ефект включення даних про IP адреси хостів у випадку застосування моделей часових рядів перевіряється на основі рівнів значущості кожної ознаки трафіку та їхнього впливу на байєсівський інформаційний критерій (BIC) і значення інформаційного критерію Akaike (AIC).

Модель ARIMA складається з рівняння прогнозування, яке може містити попередні відставання у часовому ряді, або „авторегресійні” терми, і відставання помилок прогнозу, або „ковзне середнє”. Часовий ряд повинен бути диференційованим і забезпечувати вимоги стаціонарності.

Модель ARIMA в загальному випадку можна описати у вигляді:

$$(p, d, q) \times (P, D, Q)_s \quad (2.5)$$

де p – порядок у процесу авторегресії;

d – порядок диференціювання;

q – порядок процесу ковзного середнього.

P – порядок сезонної складової у процесі авторегресії;

D – порядок диференціювання сезонної складової;

Q – порядок сезонної складової процесу ковзного середнього;

s – тривалість періоду сезонної складової.

Як видно з формули (2.5) тріада (p, d, q) - представляє собою несезонну складову моделі, а $(P, D, Q)_s$ – сезонна складова моделі ARIMA.

Автокореляційна функція АСФ визначає кореляцію між значеннями в кожній точці ряду та значеннями із затримками до цієї точки [4, 6, 5]. Ця інформація надалі використовується для обчислення частково автокореляційної функції PACF, що залишається між кожною точкою у ряді та затримками у ряді після того, як були видалені впливи всіх ближчих відставань.

Байєсівський інформаційний критерій (BIC) обчислюється за формулою

$$BIC = -2 \ln(L) + \ln(n) \cdot k \quad (2.6)$$

де L – функція правдоподібності (функція ймовірності на основі лишків із моделі);

n – об'єм навчальної вибірки;

k - кількість вільних параметрів.

Отже, задача вибору моделі на основі BIC полягає у тому, щоб включити у модель мінімум параметрів, які, при цьому, вносять найбільший вклад у значення функції правдоподібності.

Інформаційний критерій Акайке (AIC) є відносною мірою якості статистичної моделі, і в загальному випадку обчислюється як

$$AIC = 2k - 2\ln(L) \quad (2.7)$$

Інтерпретація параметрів k та L аналогічна до тієї, яка описана у формулі (2.6).

Припускаючи, що $Y_t, t = 1, 2, 3, \dots, n$ – часовий ряд, використовується параметричний підхід у відповідності до моделі прогнозування. Авторегресійні моделі з ковзним середнім (ARMA) виражаються наступним чином:

$$y_t = \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \dots + \varphi_p y_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q} \quad (2.8)$$

або еквівалентно

$$\Phi(B)y_t + \Theta(B)a_t \quad (2.9)$$

де $\Phi(B) = 1 - \varphi_1 B - \dots - \varphi_p B^p$;

$\Theta(B) = 1 - \theta_1 B - \dots - \theta_q B^q$;

p та q ранги авторегресії та ковзного середнього відповідно на стаціонарній випадковій складовій (лишки).

Моделі ARMA також можна представити у вигляді наступної формули:

$$ARMA(p, q) \quad (2.10)$$

Основне припущення при оцінці коефіцієнтів ARMA полягає в тому, що вони стаціонарні, тобто тенденція чи сезонність не можуть впливати на дисперсію. Це також означає, що дані повинні мати постійне середнє значення, дисперсію та автокореляцію в часі, що, як правило, на практиці не можливо забезпечити.

Таким чином, моделі авторегресії, інтегровані з ковзним середнім (ARIMA) виражаються як:

$$\Phi(B)(1 - b)^d y_t = \Theta(B)a_t \quad (2.11)$$

де d – кількість звичайних різниць.

Тому модель можна виразити, як $ARIMA = ARIMA(p; q; d)$.

Причиною вибору вищевказаної моделі є те, що заздалегідь було відомо, що в зібраних даних сезонний тренд відсутній.

Для кожної IP-адреси послідовність пакетів, які були надіслані та отримані (надалі позначаються як P_s та P_r відповідно), відображені у моделі ARMA/ARIMA. Природа цих послідовностей дозволяє припустити, що між даними не існує кореляції. На рис. 2.1 представлений графік послідовності (ряду) P_s , де сезонність відсутня.

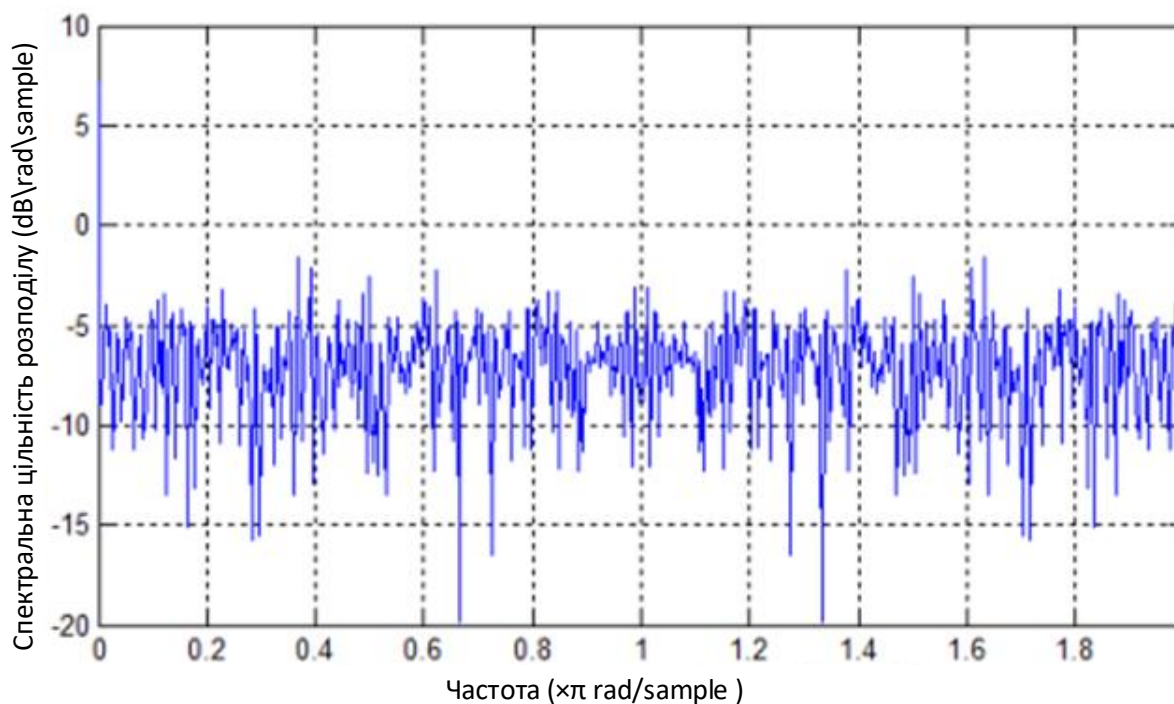


Рис. 2.1. Графік послідовності P_s з IP-адреси 195.251.240.32

Графік послідовності P_r подібний до того, який наведений на рис. 2.1.

Результати, одержані при аналізі навантаження на хости з іншими IP адресами, також не містили сезонної складової, тобто серії виявилися некорельованими (часовий ряд білого шуму) за умови, що значення часових рядів одержували у неперервних інтервалах часу. Отже, можна дослідити ряди P_s та P_r на значно менших моментах часу $n = 1000$, значення трафіку у яких позначають приймаються як $X_i, i = 1, \dots, 1000$. З точки зору повноти даних, для прогнозування часових рядів кількість у 1000 його елементів, є достатнім вхідним набором даних [7,8].

Застосування моделі ARMA/ARIMA для прогнозування часових рядів вимагає спочатку побудови графіків автокореляції та часткової кореляції – часто використовуваного інструменту для ідентифікації залежностей у моделях ARMA / ARIMA для виявлення стаціонарності даних. Крім того, часткові графіки автокореляцій корисні для ідентифікації порядку p моделі авторегресії, а графіки автокореляцій – для ідентифікації порядок q моделі ковзного середнього.

Діаграми автокореляції X_i показали, що ані сезонність, ані тренд не є присутніми у часовому ряді, однак наявними є загальні ознаки нестационарності у даних. Використовуючи різниці першого порядку, найпоширенішими та найкраще обґрунтованими у моделях часових рядів P_s та P_r є ARIMA (1; 1; 1) $y_t - (1 + \varphi)y_{t-1} + \varphi y_{t-2} = a_t - \theta a_{t-1}$, $\varphi = 0.676$, $\theta = 0.751$ та ARIMA (0; 1; 1) $y_t - y_{t-1} = a_t - \theta a_{t-1}$, $\theta = 0.564$.

Варто зазначити, що кілька моделей добре підходили до одного і того ж набору даних, і для того, щоб обрати найкращі застосовується інформаційний критерій Акайке (AIC) [9] та інформаційний критерій Байєса (BIC) [12].

На рис. 2.2 показано перші 1000 елементів початкової серії P_s разом із двома різними моделями ARIMA, які забезпечили найкраще прогнозування для заданого набору даних. По аналогії, на рис. 2.3 показано найкраще визначені моделі для інших елементів (1001-4000), у неперервних частинах 1000 елементів серії P_s – значення трафіку за IP-адресою 195.251.240.32.

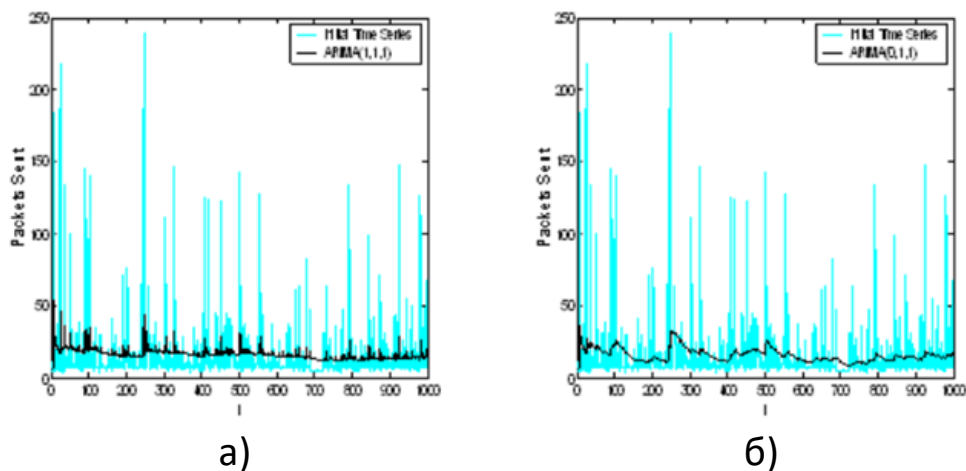


Рис. 2.2. Дві найкращі моделі прогнозування:

а) ARIMA (1,1,1) та б) ARIMA (0,1,1)

На рис. 2.1 показано перших 1000 елементів ряду одержаних пакетів P_s на вузлі з IP-адресою 195.251.240.32.

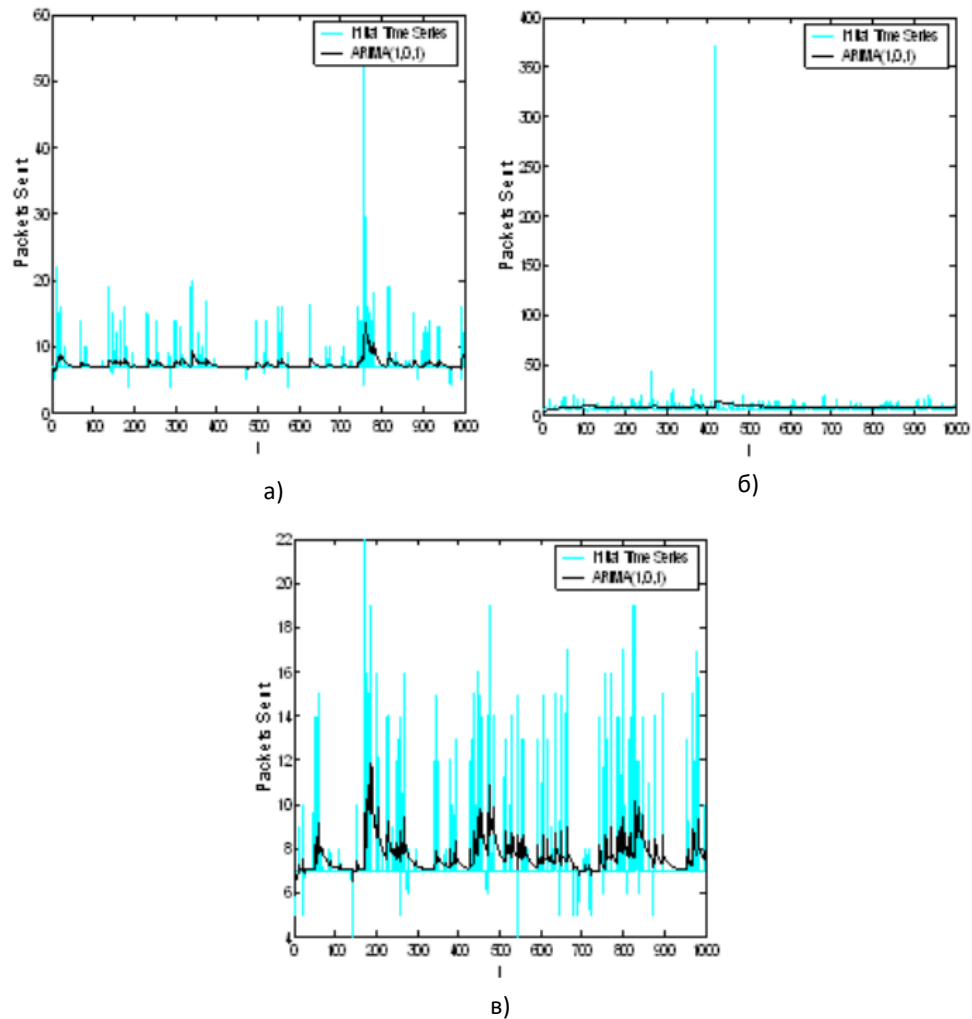


Рис. 2.3. Графіки прогнозування трафіку моделі ARIMA
 а) інтервал 1001-2000 елементів, б) інтервал 2001-3000 елементів,
 в) 3001-4000 елементів

На рис.2.3 наведено графіки щодо прогнозування значень одержаного трафіку у вузлі з IP-адресою 195.251.240.32 на основі моделі ARIMA (1,0,1) (а) на інтервалі від 1001-2000 елементів, ARIMA (1,0,1) (б) на інтервалі 2001-3000 елементів та ARIMA (1,1,1) (в) на інтервалі від 3001 до 4000 елементів.

У табл. 2.1 наведено параметри моделей ARIMA для перших чотирьох частин з 1000 елементів надісланих пакетів даних з IP-адреси 195.251.240.32.

Таблиця 2.1

Параметри моделей ARIMA (вихідний трафік)

P_s^*	Моделі прогнозування	φ	θ
1-1000	ARIMA(0; 1; 1)	0.967	0.875
1-1000	ARIMA(1; 1; 1)	0.986	0.870
1001-2000	ARIMA(1; 0; 1)	0.998	0.896
2001-3000	ARIMA(1; 0; 1)	0.998	0.896
3001-4000	ARIMA(1; 1; 1)	0.109	0.940

* P_s – ряд, сформований на основі надсилання пакетів вузлом комп'ютерної мережі

Як видно з табл. 2.1, цілком можливо, що для певних підмножин даних може використовуватись кілька моделей. Зокрема, для перших 1000 елементів як модель ARIMA (0; 1; 1), так і моделі ARIMA (1; 1; 1) виконують прогнозування досить точно. По аналогії дані моделі ARIMA можна застосувати і для прогнозування одержаних пакетів даних P_r за IP-адресою 195.251.240.32.

У табл. 2.2 наведено моделі ARIMA з відповідними оцінками параметрів для одержаних даних (вхідного трафіку) за тією ж адресою.

Таблиця 2.2

Параметри моделей ARIMA(вхідний трафік)

P_r^{**}	Моделі прогнозування	φ	θ
1-1000	ARIMA(1; 1; 1)	0,634	0,960
1001-2000	ARIMA(1; 0; 1)	0,999	0,937
2001-3000	ARIMA(1; 0; 1)	0,949	0,620
3001-4000	ARIMA(0; 1; 1)	0,000	0,912

** P_r – ряд, сформований на основі одержання пакетів вузлом комп'ютерної мережі

За таким самим принципом виконується прогнозування рядів P_s та P_r для інших вузлів комп'ютерної мережі. Згідно з одержаними результатами, які наведені у табл. 2.1 і табл. 2.2 можна зробити висновок про те, що оптимальним варіантом прогнозування на даному наборі даних є модель ARMA (p; q) або ARIMA (p; q; d), які враховують особливості та параметри процесів авторегресії та ковзного середнього.

Варто також відмітити, що така процедура є ще більш оптимальною, коли ряд (трафік) розбити на менші частини, у порівнянні, коли прогнозування виконувати на всьому наборі даних. Такий підхід є ефективним при виконанні коротко- та середньо- строкових прогнозів.

2.2.2. Прогнозування трафіку комп'ютерної мережі на основі підходу deep learning

У сучасну епоху світ Інтернету стикається із зростаючою проблемою щодо розробки більш надійних та стійких стратегій мережевої комунікації. Швидкі зміни у топологіях мереж стимулюють значні зміни у циркуляції трафіку. Важливу роль при цьому відіграють Інтернет-провайдери (ISP) – організації, що надають послуги доступу до мережі Інтернет. На рис. 2.4 наведено загальну інфраструктуру ISP.

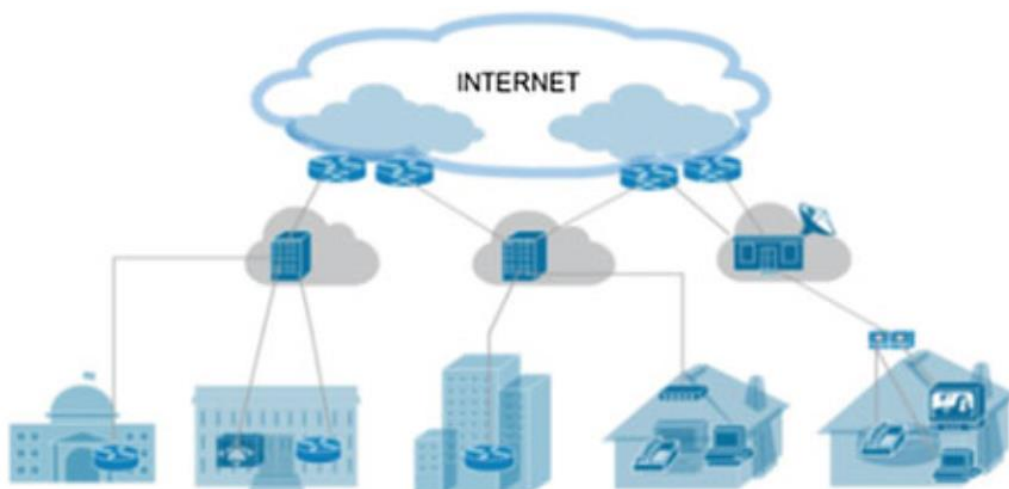


Рис. 2.4. Базова інфраструктура інтернет-провайдера (ISP)

Якість обслуговування (QoS) представляє собою концепцію, при якій частоту помилок, швидкість передачі та інші фізичні характеристики мережі можна виміряти, покращити та, певною мірою, спрогнозувати гарантоздатність. Аналіз навантаження мережевого трафіку відіграє важливу роль при оптимізації ресурсів, проектуванні, управлінні та контролі мережевих послуг [10]. ATM і високошвидкісні мережі надають інтегровані послуги щодо передачі та опрацювання голосових, відео та інших видів інформації. Ці типи послуг потребують адекватного механізму контролю черг у мережі, зберігаючи при цьому максимальне використання мережі, оптимізуючи розподіл ресурсів таким чином, щоб надання послуг імітувало обмеження якості обслуговування. Тому прогнозування Інтернет-трафіку є дуже важливим для розвитку більш функціональних засобів організації трафіку та виявлення аномалій. З іншого боку, моделі, розроблені для майбутнього прогнозування трафіку, підтримують виявлення аномалій у мережах передачі даних. Більше того, нерегулярну кількість елементів спаму та інших атак безпеки можна виявити, порівнявши фактичний трафік із прогнозом відповідних моделей.

Моделювання трафіку є фундаментальним аспектом при оцінюванні продуктивності мережі. Дослідження трафіку і навантаження на вузли мережі продемонстрували їхню нелінійну природу [12, 13]. Трафік представляють у вигляді часових рядів з нелінійними та хаотичними характеристиками [14, 15]. Встановлено, що ознаки трафіку у часових рядах можуть бути корельованими як на короткому, так і на довгому проміжках часу. Отже, більшість важливих кореляційних зв'язків та дискретних властивостей часто ігнорують при застосуванні простих моделей. Тому пропонується модель в основі якої лежить підхід глибокої штучної нейронної мережі (ANN).

Для прогнозування трафіку пропонується скористатись підходом аналізу часових рядів, використовуючи основні атрибути глибокого навчання та його важливі параметри для визначених архітектурних топологій мереж.

Модель для прогнозування трафіку будується на основі стекінгу моделей Restricted Boltzmann Machines (RBM) при створенні глибокої архітектури

нейронної мережі. Шар RBM спочатку навчається без учителя. Кожен шар визначає нелінійне представлення даних. Абстракції верхнього прихованого шару формують нелінійні ієрархічні функції у вигляді набору вхідних даних для прогнозування навантаження мережевого трафіку. Таким чином, прогнозування навантаження трафіку виконується у вихідному шарі, який тренується під контролем вчителем на етапі «тонкого» налаштування всієї моделі.

Використання ANN інкапсульовано в широкому діапазоні застосувань на підставі його узагальнюючої здатності до непередбачених ситуацій. ANN навчається на наборі вхідних даних для досягнення конкретного цільового результату за допомогою відповідного алгоритму навчання, поки результати мережі, як правило, не відповідають заданим цілям.

Таким чином вибираються параметри мережі, які можуть вивести нелінійну залежність між заданим входами та виходами нейромережі. До цієї міри навчена модель оцінює прогнози щодо невидимих вхідних зразків.

Для того, щоб збільшити ефективність ANN, його обчислювальна здатність вимагає більшої кількості нейронів на шар або збільшення кількості прихованих шарів. Це призводить до глибокої архітектури нейронної обчислювальної моделі, в якій приховані шари виконують функцію декількох абстрактних шарів.

Що стосується глибоких архітектур ANN, то проводяться масові дослідження. Глибокі ANN видають кращі результати у порівнянні з неглибокими архітектурами ANN [16]. Глибокі ANN містять багато рівнів нелінійності залежно від глибини прихованих шарів. Глибока ієрархічна архітектура дозволяє їм ефективно представляти сильно нелінійні зв'язки та дуже різні функціональні абстракції. Хоча було незрозуміло, як тренувати такі глибокі мережі, як випадкова ініціалізація мережевих параметрів часто застряє в поганих рішеннях [17].

У [18] введено «жадібний» алгоритм навчання, який дозволяє тренувати послідовно кожен шар нейронної мережі. Глибокі архітектури можуть бути сформовані або шляхом створення автоенкодерів, або Deep Belief Networks (DBN) RBM [18]. DBN є найпоширенішим та найефективнішим підходом серед усіх моделей глибокого навчання. DBN утворюється шляхом стекінгу RBM. RBM – це

двостороння графова, імовірнісна модель, заснована на енергії, що трактується як стохастична нейронна мережа. RBM базується на двошаровій структурі, до складу якої входять видимі і приховані вершини, як показано на рис. 2.5.

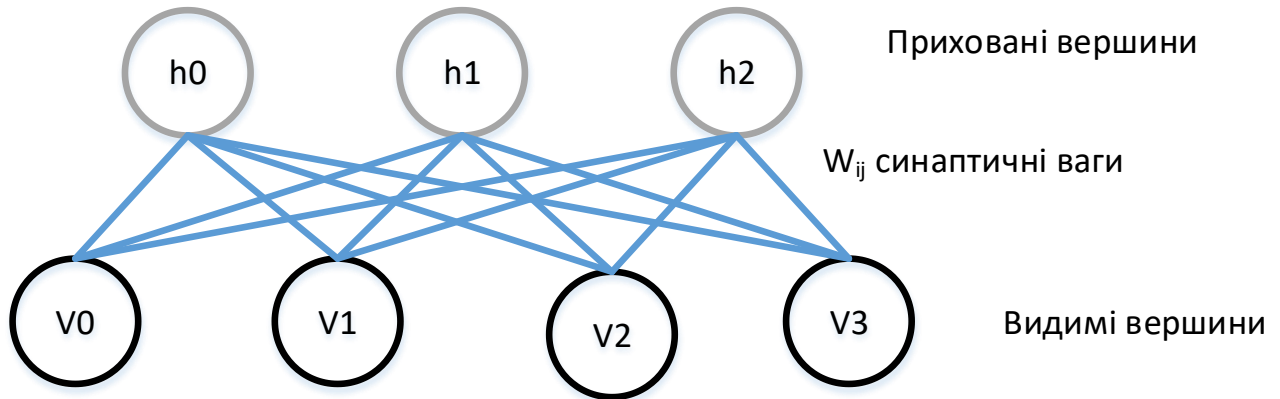


Рис. 2.5. Структура RBM

Вершини графової структури є бінарними, тобто h і $v \in \{0,1\}$ та умовно незалежними одна від одної. Отже, імовірнісне представлення цих прихованих і видимих вершин з сигмоподібною функцією активації $\sigma(x) = \frac{1}{1+e^{-x}}$ можна представити у вигляді:

$$p(h_{i=1}|v) = \sigma(\sum_{j=1}^m w_{ij}v_j + c_i) \quad (2.12)$$

$$p(V_{j=1}|v) = \sigma(\sum_{i=1}^m w_{ij}h_i + b_j) \quad (2.13)$$

де w_{ij} – це синаптична вага ребра, що з'єднує вершини V_j та h_i ;

b_j та c_i – дійсні відхилення, пов'язані з j -ою видимою та i -ою прихованою змінною відповідно. Вибірка в RBM отримується шляхом запуску ланцюга Маркова з використанням збіжності ряду та відбору даних за Гіббсом.

Значення вагових параметрів оновлюється зі швидкістю зміни, як показано у формулі 2.14.

$$\Delta w_{ij=1} = \epsilon (\langle v_j h_i \rangle_{data} - \langle v_j h_i \rangle_{model}) \quad (2.14)$$

Для прогнозування трафіку у комп'ютерних мережах запропоновано використати підхід deep learning, що використовується як основна модель при виконанні прогнозу. Для реалізації моделі deep learning пропонується використати три архітектури для прогнозування часового ряду, який представляє трафік у комп'ютерній мережі. Загальна архітектура вхідних / вихідних змінних та структура прихованих шарів глибокого ANN для розв'язку задачі прогнозування показана на рис. 2.6.

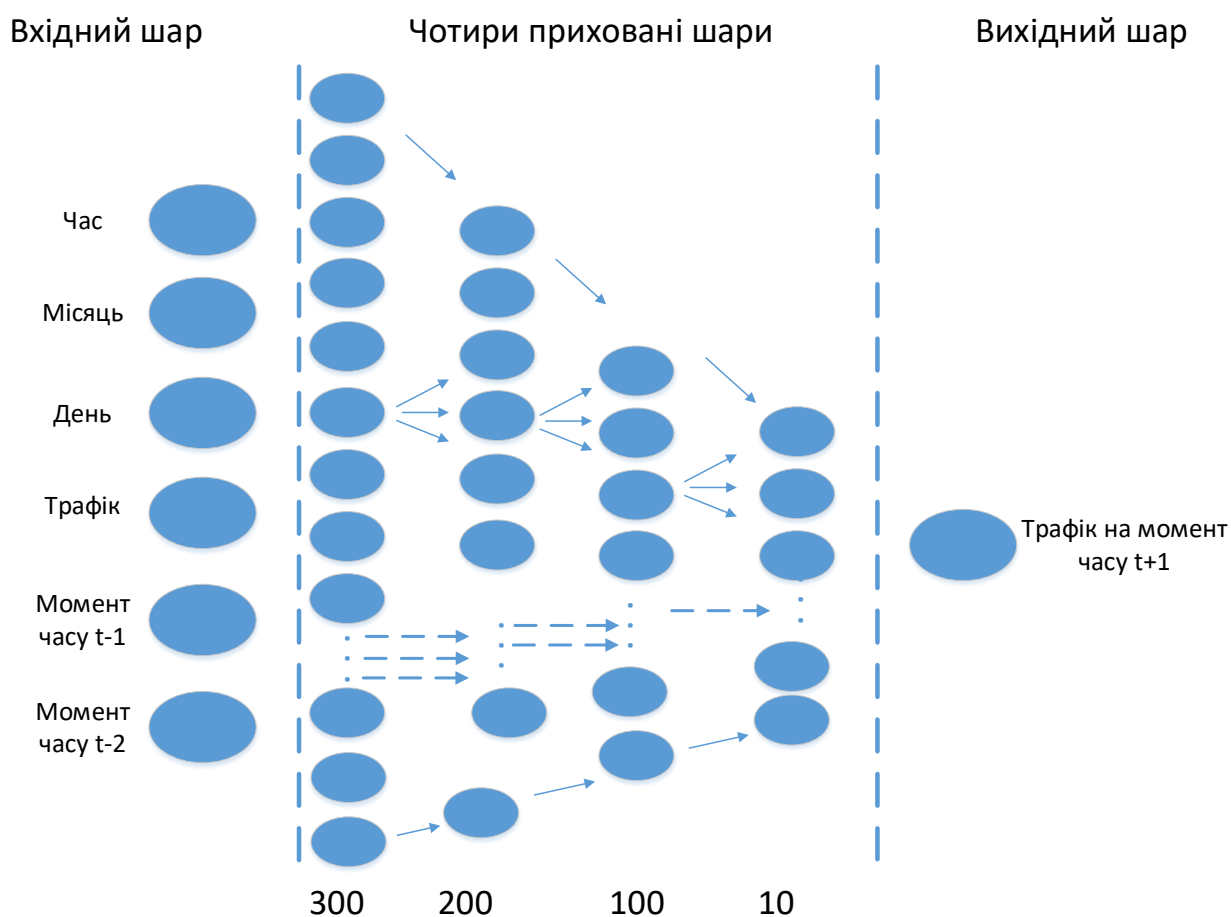


Рис. 2.6. Архітектура глибокої нейронної мережі

У даному випадку глибока нейронна мережа використовується для прогнозування значень трафіку на 1 год вперед, тобто коротко-строкове прогнозування. Глибока ієрархія моделі перетворює набір вхідних ознак у

нелінійні абстракції вищого рівня для визначення більш важливих моделей та тенденцій, доступних у наборі даних.

Як видно з рис. 2.6, модель реалізована на основі DBN з 4-ма прихованими шарами розміром 300–200–100–10, вхідним шаром, що містить 6 нейронів для вхідних змінних та вихідним шаром, який складається з одного нейрону для прогнозування майбутнього навантаження трафіку. Не можна довільно вибирати топологію мережі, тобто глибину мережі та розмір прихованого шару. Учені, зокрема у [19], виділили три різні топології для побудови моделі, визначаючи вибір кількості прихованих шарів. Це говорить про те, що кращий вибір топології моделі полягає у дотриманні цих показників моделі, яка є постійною по всій моделі, або відповідає порядку збільшення/зменшення розміру. Отже, архітектура моделі базується на зменшенні розміру прихованого шару знизу вгору. Кількість нейронів у першому прихованому шарі розраховується за допомогою моделювання на основі методу Монте-Карло з діапазону (100–600).

Кожен шар DBN самостійно навчається як RBM з функцією активації сигмоїдної форми. Оптимальні налаштування для DBN показані в таблиці 2.3.

Таблиця 2.3

Параметри моделі для прогнозування трафіку

Параметри	Значення
Максимальні кількість ітерацій	100
Початкове значення інтервалу ініціалізації синаптичних ваг	0.5
Кінцеве значення інтервалу ініціалізації синаптичних ваг	0.9
Швидкість навчання	0.1
Розмір пакету даних (batch size)	5
Кількість викидів (dropout rate)	0.0
Функція активації	Сигмоїд

Спочатку перший прихований шар, що знаходиться одразу із вхідним шаром тренується без учителя, формуючи початковий набір значень параметрів для першого шару нейронної мережі. Після цього виходи, отримані з цього шару, використовуються як новий набір функцій, які подаються на вхід для наступного прихованого шару. Шари додаються послідовно і тренуються самостійно та незалежно.

Ця процедура відома як жадібна попередня підготовка шару. Процедура повторюється довільну кількість разів, щоб отримати більше нелінійних залежностей. Це забезпечує ініціалізацію параметрів нейронної мережі. Після того, як шари RBM в DBN пройшли навчання, додається вихідний шар із функцією втрат. Таким чином, мережа глобально налаштована за допомогою керованого алгоритму прогнозування цільового значення. Загальна кількість ітерацій для навчання мережі становить 500.

2.3. Висновки до розділу

У даному розділі одержано наступні наукові та практичні результати:

1. Проаналізовано підхід для прогнозування часових рядів на основі методів екстраполяції, що дало змогу обґрунтувати ефективність їхнього застосування, зокрема методу ковзного середнього, для побудови прогнозів у випадку, коли трафік у мережі є стабільним та регулярним у часі.

2. Досліджено методи регресійного аналізу в контексті прогнозування трафіку у комп'ютерній мережі, у результаті якого встановлено, що він є найбільш використовуваним на практиці і базується на визначенні коефіцієнтів регресії та оцінюванні точності прогнозу, при цьому припускають, що коефіцієнти регресії є сталими.

3. Запропоновано застосування моделей ARMA та APIMA для прогнозування часових рядів при аналізі трафіку локальних комп'ютерних мереж, коли трафік передачі та приймання пакетів є однорідним і взаємодія між вузлами мережі приблизно рівномірна, що дало змогу забезпечити високу точність

прогнозування трафіку у коротко і середньо-строковому інтервалах часу на рівні $\geq 95\%$ без врахування сезонної і трендової складової.

4. Запропоновано та спроектовано архітектури глибоких нейронних мереж, вхідний шар яких містить 7 нейронів, кількість проміжних шарів становить 4 рівні, вихідний шар містить один нейрон, що дало змогу практично в реальному часі забезпечити високу точність прогнозування (97%) трафіку для інтернет-сервіс провайдерів на інтервал 1 год вперед.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ЗАПРОПОНОВАНИХ МОДЕЛЕЙ ТА МЕТОДУ ПРОГНОЗУВАННЯ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ

3.1. Аналіз вимог, структури набору вхідних даних та проектування архітектури програмного засобу для прогнозування трафіку у комп'ютерних мережах

Для реалізації моделі прогнозування трафіку у комп'ютерних мережах необхідно визначити вимоги до програмного забезпечення, проаналізувати набори вхідних даних, спроектувати та імплементувати модель прогнозу.

Для моделювання та прогнозування трафіку у комп'ютерній мережі, в якості експерименту, взято відкриті дані з ресурсу <https://www.kaggle.com/crawford/computer-network-traffic>.

Структура вхідного набору даних містить 21 тис. рядків із даними за три місяці для різних реальних мереж. Зокрема, було використано 10 IP адрес локальних комп'ютерів, які генерували трафік протягом вказаного вище періоду часу.

Структура data set (набору даних) включає наступні поля:

- date – поле дата, яке представляється у форматі «уууу-мм-дд», область визначення якого лежить в інтервалі [“2006-07-01”; “2006-09-30”];
- l_ipn – локальна IP адреса, яка закодована в діапазоні від 0 до 9;
- r_asn – віддалені ASN, які ідентифікують віддалених інтернет провайдерів (ISP);
- f – потік з'єднань, що представляє кількість з'єднань за конкретний день (дату).

На рис. 3.1 наведено фрагмент вхідного набору даних.

date	l_ipn	r_asn	f
7/1/2006	0	701	1
7/1/2006	0	714	1
7/1/2006	0	1239	1
7/1/2006	0	1680	1
7/1/2006	0	2514	1
7/1/2006	0	3320	1
7/1/2006	0	3561	13
7/1/2006	0	4134	3
7/1/2006	0	5617	2
7/1/2006	0	6478	1
7/1/2006	0	6713	1
7/1/2006	0	7132	1
7/1/2006	0	9105	1
7/1/2006	0	10738	1
7/1/2006	0	10994	1
7/1/2006	0	12334	1
7/1/2006	0	12524	1
7/1/2006	0	12542	1
7/1/2006	0	13343	1
7/1/2006	0	13446	1
7/1/2006	0	13462	20
7/1/2006	0	13867	1
7/1/2006	0	13989	1

Рис. 3.1. Вхідний набір даних для прогнозування трафіку комп'ютерної мережі

Для програмної реалізації моделі прогнозування трафіку необхідно забезпечити виконання функціональних вимог, які представлені у вигляді use case діаграми на рис. 3.2.

Виконання функціональної вимоги щодо налаштування гіперпараметрів моделі на основі глибокого навчання передбачає:

- встановлення значення розміру пакету даних з основної вибірки (batch size);
- налаштування максимальної кількості ітерацій;
- розмір пакету даних, які не будуть включатись в batch size;
- мінімальне значення ініціалізації синаптичних ваг;
- максимальне значення при ініціалізації синаптичних ваг;
- швидкість навчання моделі;
- функція активації.

Окрім цього, для оцінювання результатів прогнозування передбачає застосування наборів функціоналів якості, зокрема:

- MSE (Mean Square Error);
- RMSE (Root Mean Square Error);
- MAPE (Mean Absolute Percentage Error);
- R^2 .

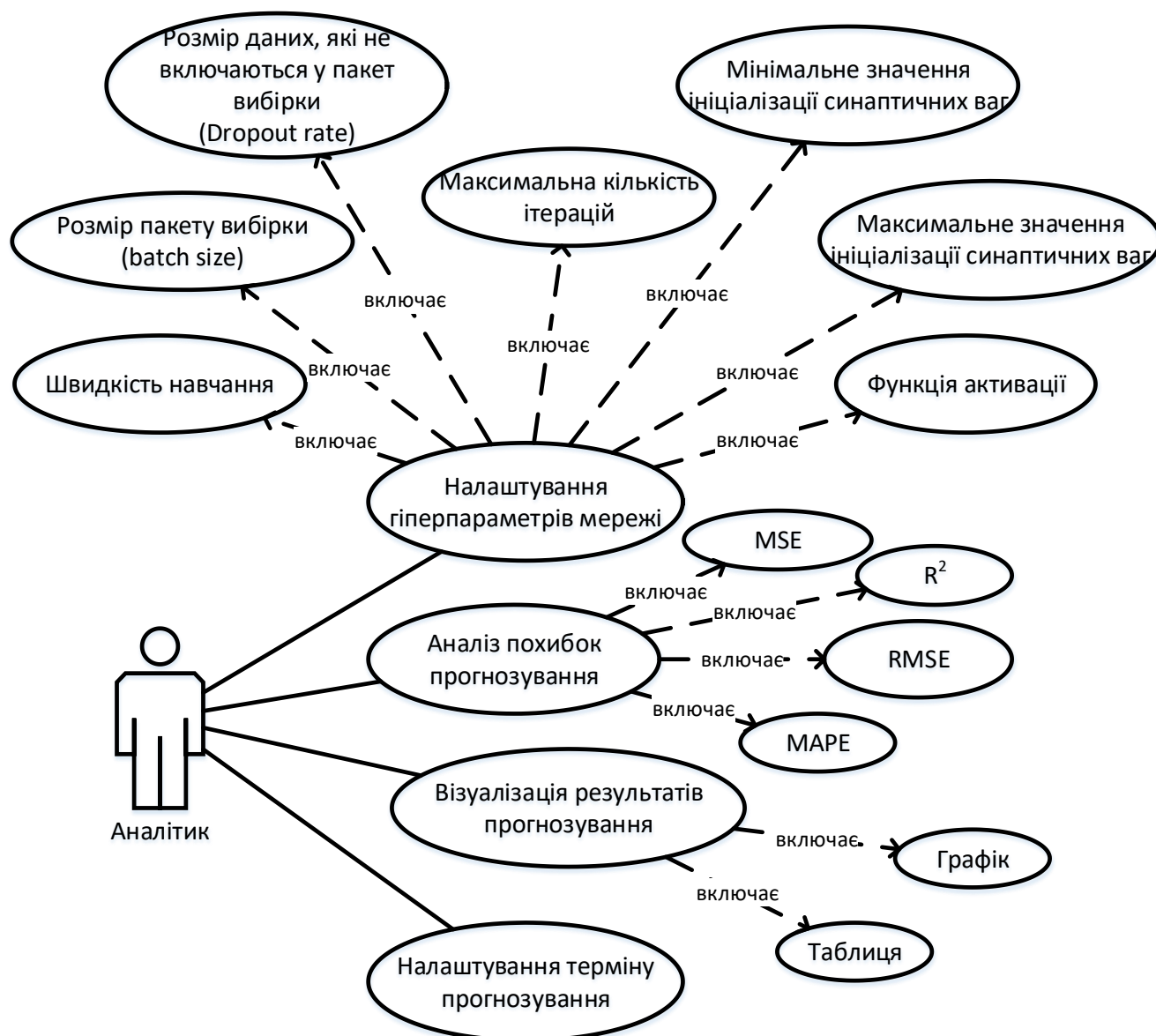


Рис. 3.2. Вимоги до програмної реалізації моделі прогнозування

Візуалізація прогнозованих даних відбувається у табличній формі та у вигляді графіків. Окрім цього, аналітик повинен мати можливість налаштувати період прогнозування та мати можливість завантажити вхідний набір даних (data set).

У даному випадку (рис.3.2), аналітик має повністю доступ до всіх параметрів моделі та функціоналів якості, а також до вхідних даних. З іншої сторони, для того, щоб сторонні аналітики могли використовувати побудовану модель як «чорну скриньку», вимоги до програмної моделі та її імплементації матимуть вигляд, як показано на рис. 3.3.

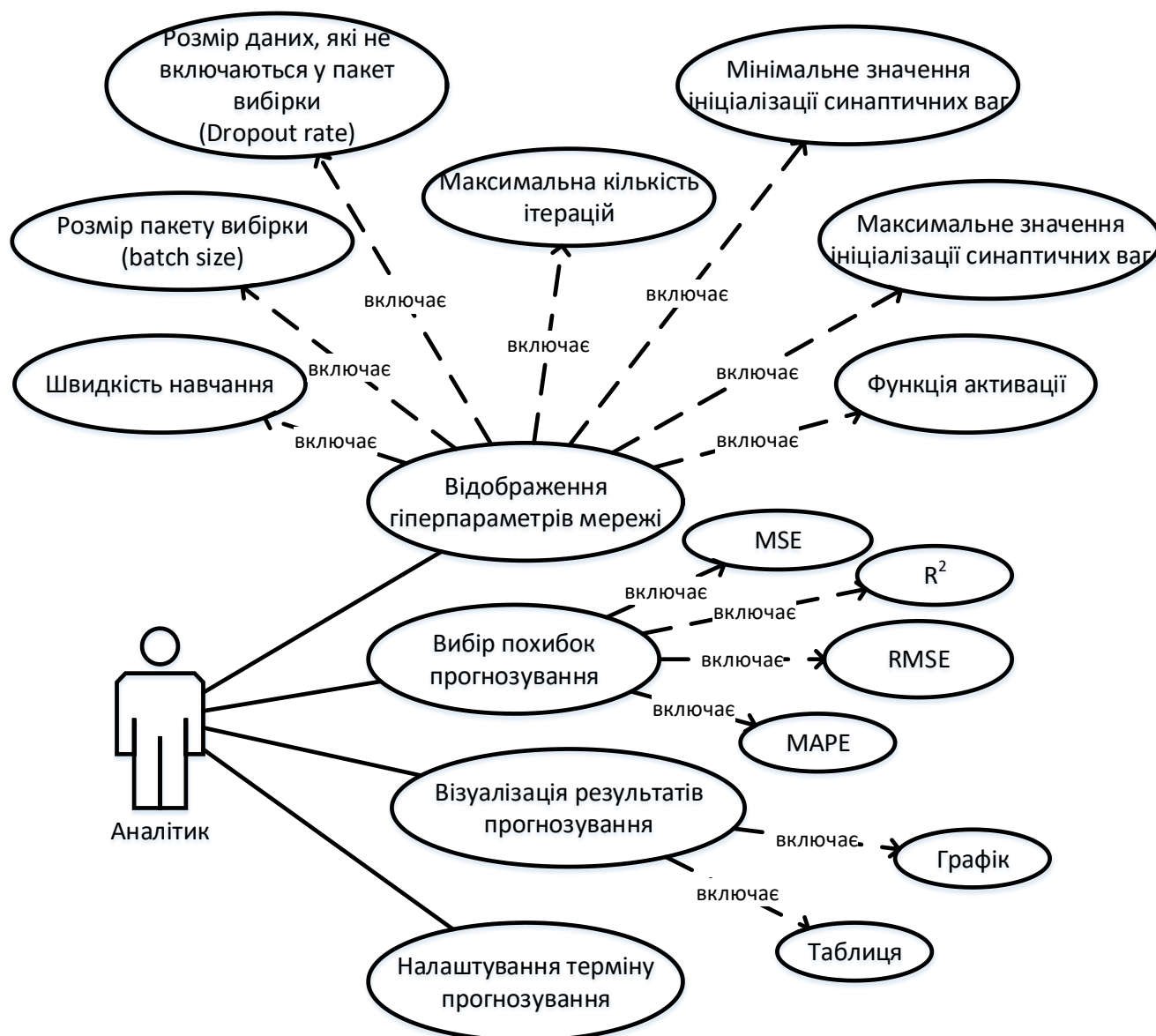


Рис. 3.3. Вимоги до ПЗ, що використовують сторонні аналітики

Як показано на рис. 3.3, сторонні аналітики можуть лише налаштувати параметр терміну прогнозування, а змінювати параметри моделі чи додавати функціонали якості вони не можуть. Для вирішення та забезпечення виконання функціональних вимог, приведених на рис. 4.3, пропонується скористатися підходом контейнеризації, наприклад, docker container.

У даному випадку, можливе підключення багатьох сервісів для роботи з побудованою моделлю. Однак для ефективності її застосування необхідно запакувати її у так званий контейнер і забезпечити доступ до нього.

Оскільки, прогнозування трафіку у комп'ютерних мережах, запропоновано вирішувати на основі нейромережових моделей, то для початку необхідно побудувати їхні архітектури.

3.2. Проектування архітектури та імплементація моделі прогнозування трафіку з використанням мови програмування Python

Програмне забезпечення, в основі якого лежить, спроектована та реалізована модель прогнозування трафіку комп'ютерних мереж повинна відповідати архітектурі, як наведено на рис. 3.4.

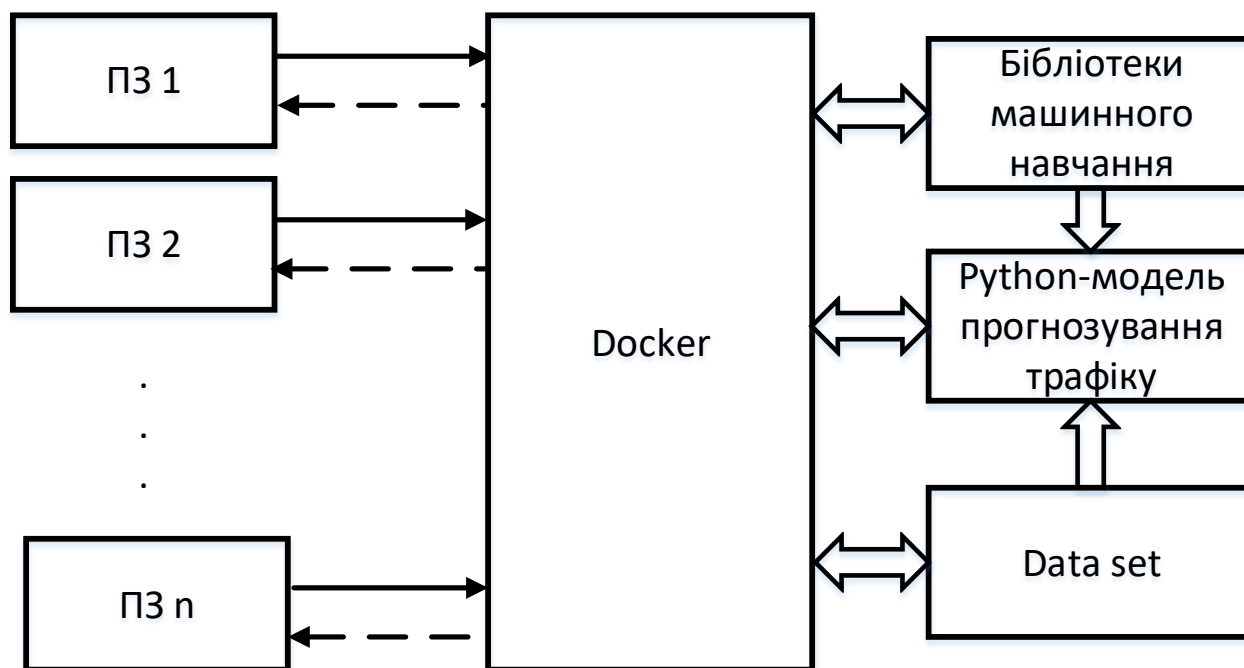


Рис. 3.4. Архітектура програмного засобу прогнозування трафіку у комп'ютерних мережах

Як видно з рис. 3.4 модель прогнозування трафіку разом з бібліотеками машинного навчання і вхідним набором даних запаковано у Docker контейнер. Зовнішнє програмне забезпечення може використовувати спроектовану модель прогнозування і передавати власні набори вхідних даних. Перейдемо до реалізації моделі архітектури нейронної мережі та її імплементації засобами мови

програмування Python. Базову архітектуру глибокої нейронної мережі побудовано у розділі 2, однак необхідно дослідити її, шляхом зміни кількості нейронів у прихованому шарі мережі. На рис. 3.5 показано видозміни базової архітектури, зокрема топологія прихованих шарів з кількістю нейронів 300-300-300-300 (рис.3.5 (а)) та 300-200-10-3 (рис.3.5 (б)).

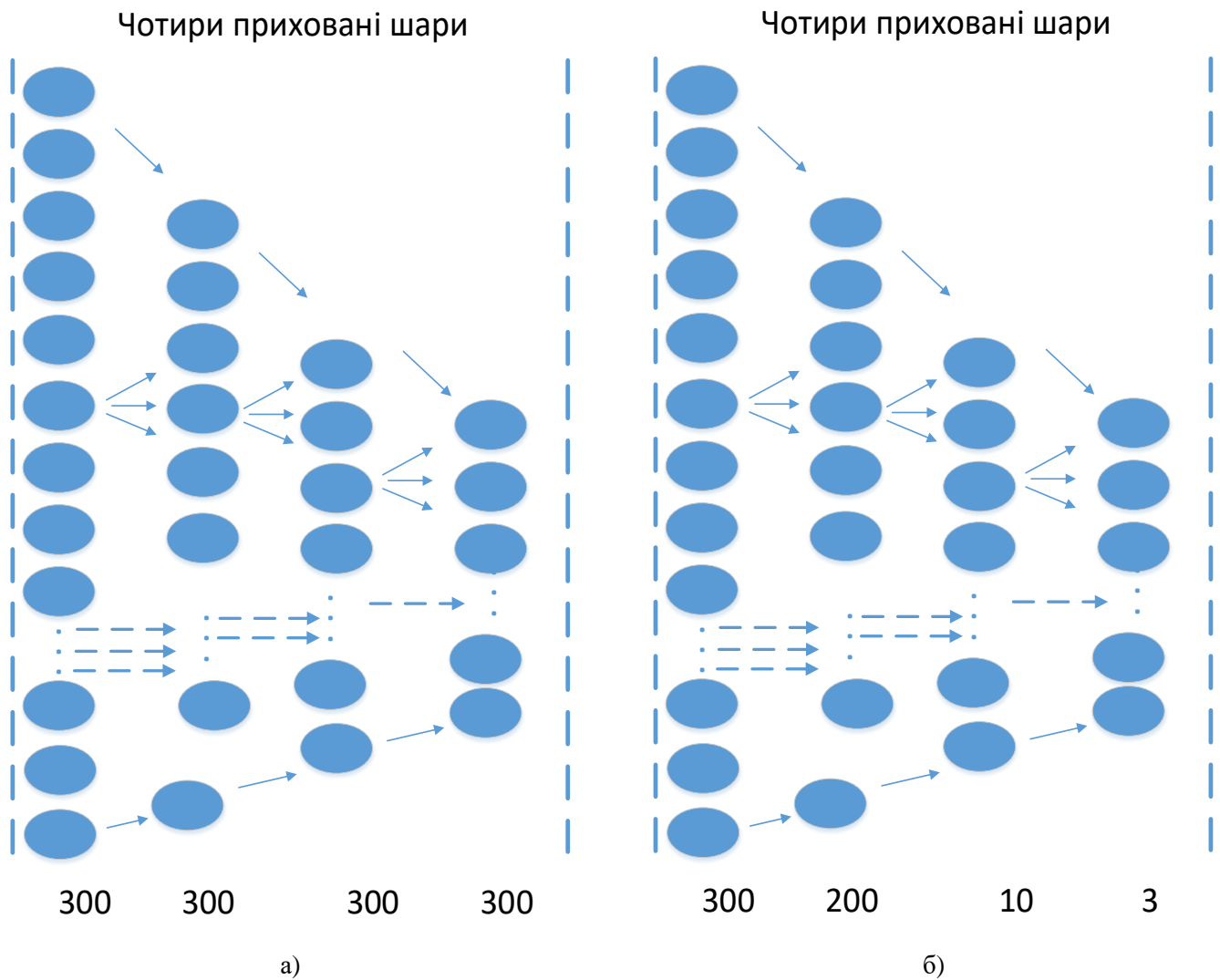


Рис. 3.5. Видозміни прихованого шару в архітектурі нейронної мережі

Дані, які необхідні для прогнозування трафіку у комп'ютерній мережі дають змогу сформувати вхідний шар нейронної мережі. Після формування вхідного шару пропонується створити 4 прихованих шари, а вихідний шар – результат прогнозування міститиме прогнозоване значення трафіку на момент часу $t+1$.

Для того, щоб зчитати дані з набору, представленого у csv форматі реалізовано скрипт, який наведено у лістингу 3.1.

Лістинг 3.1. Зчитування даних з data set

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
#Importing data
df = pd.read_csv('D:\Master\traffic.csv')
#Printing head
df.head()
```

Результат зчитування у табличному вигляді та у вигляді графіку наведено відповідно на рис. 3.6. та рис. 3.7.

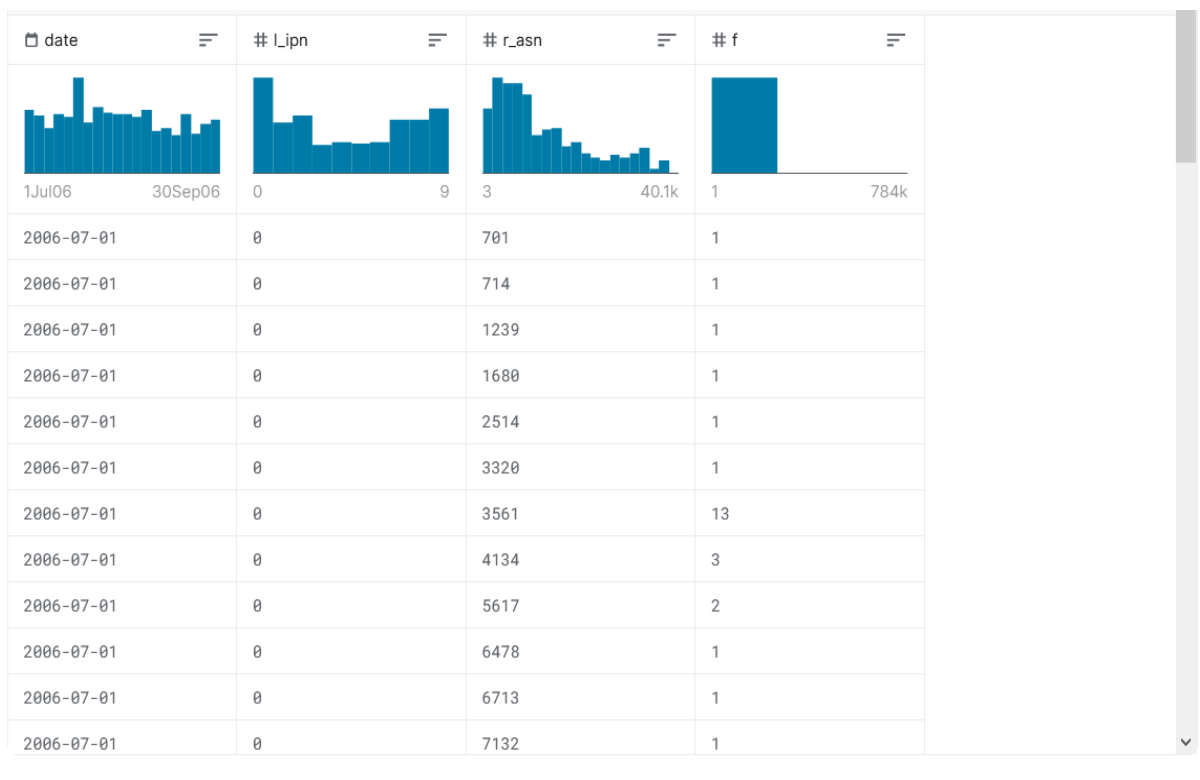


Рис. 3.6. Табличний вигляд зчитаних даних

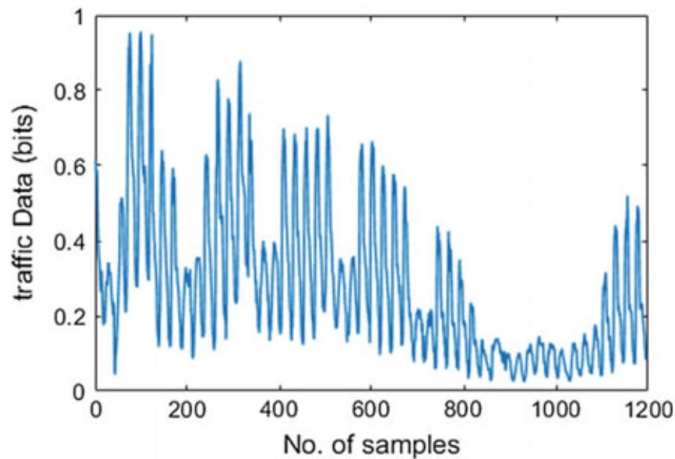


Рис. 3.7. Графічне представлення зчитаних даних, що показує залежність трафіку та кількості хостів

У лістингу 3.2 наведено імпорт бібліотек для виділення сезонної складової у часовому ряді для прогнозування трафіку у комп'ютерній мережі.

Лістинг 3.2. Лістинг для виділення сезонної складової у тренді

```
import statsmodels.api as sm
sm.tsa.seasonal_decompose(train.Count).plot()
result = sm.tsa.stattools.adfuller(train.Count)
plt.show()
```

Далі необхідно навчити алгоритм навчання для прогнозування трафіку у комп'ютерних мережах, для цього реалізовано код, що наведений у лістингу 3.3.

Лістинг 3.3. Лістинг прогнозування трафіку та візуалізації даних

```
fit1 = ExponentialSmoothing(np.asarray(train['Count']),
,seasonal_periods=7 ,trend='add', seasonal='add',).fit()
y_hat_avg['Holt_Winter'] = fit1.forecast(len(test))
plt.figure(figsize=(16,8))
plt.plot( train['Count'], label='Train')
plt.plot(test['Count'], label='Test')
plt.plot(y_hat_avg['Holt_Winter'], label='Holt_Winter')
plt.legend(loc='best')
plt.show()
```


У результаті одержано візуалізовані графіки навчання нейромережі за шарами, за властивостями моделі нейронної мережі та гістограма помилок. На рис. 3.8 наведено графік, що показує метрику RMSE (Root mean square error) середньо квадратичного відхилення за шарами при навчанні без учителя.

На рис. 3.9 наведено нелінійне представлення властивостей моделі за прихованими шарами.

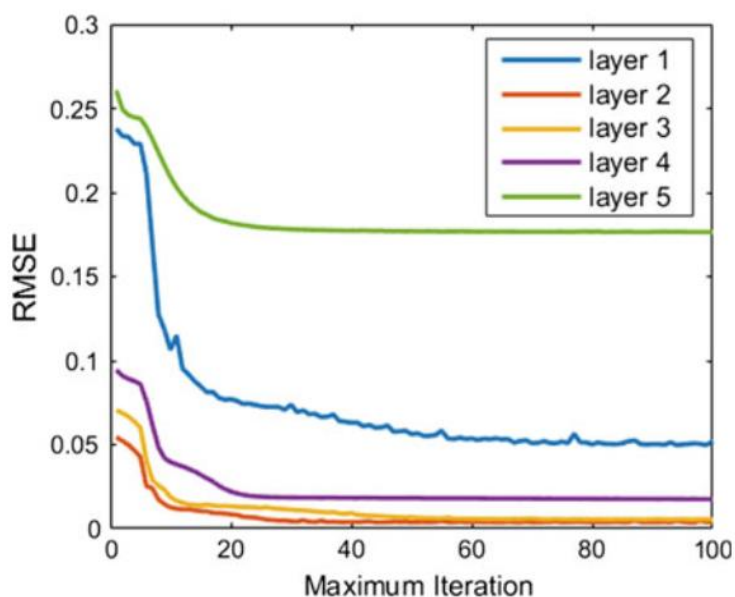


Рис. 3.8. Відхилення за шарами при навчанні без учителя

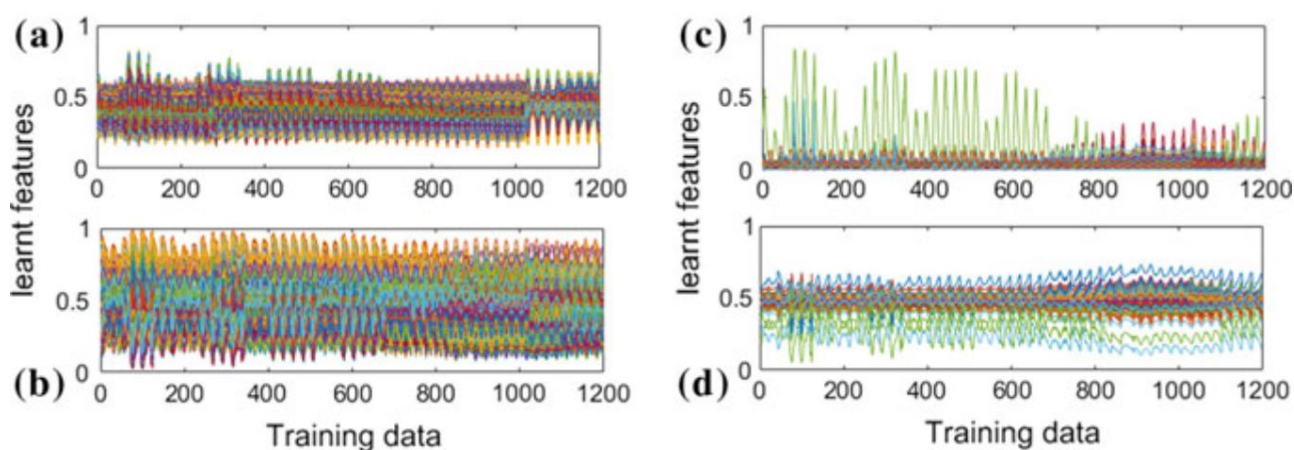


Рис. 3.9. Візуалізація властивостей моделі за прихованими шарами

Гістограма помилок для навчальної вибірка показана на рис. 3.10.

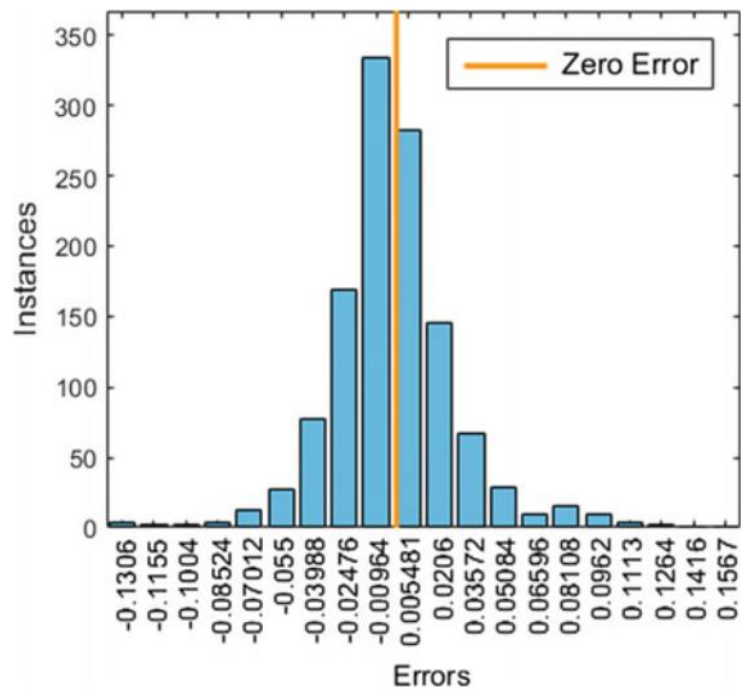


Рис. 3.10. Гістограма помилок для навчальної вибірки

У результаті застосування такої архітектури нейронних мереж було проведено тестування моделі і одержано результат, який наведено на рис. 3.11, а також у вигляді метрик якості моделі у таблиці 3.1.

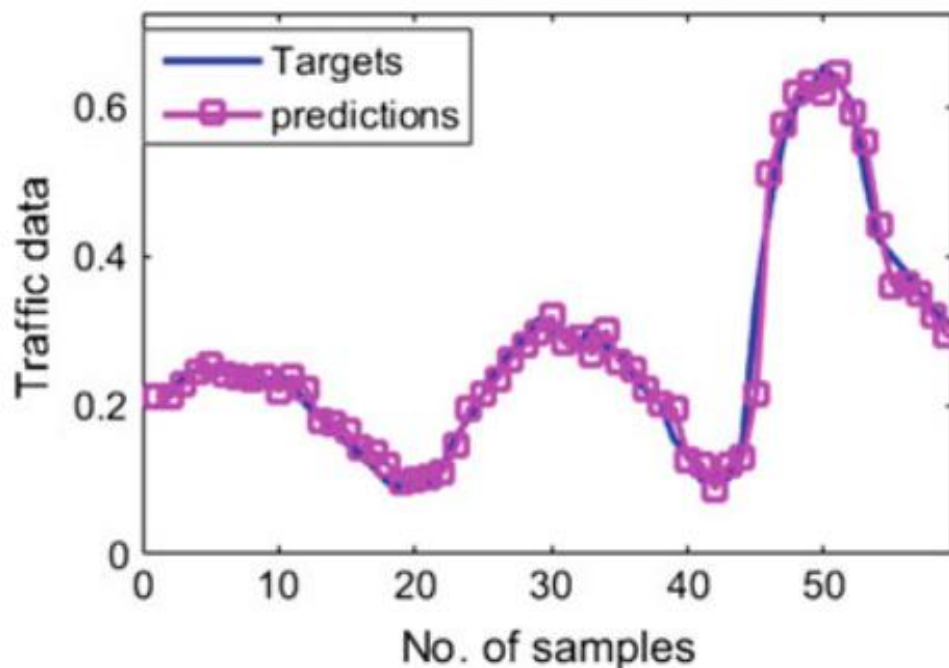


Рис. 3.11. Тестування параметрів моделі прогнозування

Результати тестування моделей прогнозування

Глибокі нейронні мережі DNN	RMSE		MSE		R ²
	Навчальна вибірка	Тестова вибірка	Навчальна вибірка	Тестова вибірка	Тестова вибірка
Модель (300-200-100-10)	0,0300	0,0286	8,9e-04	8,1e-04	0,987
Модель (300-300-300-300)	0,0319	0,0310	0,0010	9,5e-04	0,984
Модель (300-200-10-3)	0,0331	0,0427	0,0011	0,0018	0,976

Як видно з проведених результатів прогнозування, точність застосування моделі нейромережі з архітектурою ANN становить приблизно 97-98%. У додатку Б наведено більш детальні результати оцінки точності прогнозування.

3.3. Створення Docker контейнера для програмного модуля

Docker є технологією та інструментом контейнеризації з відкритим програмним кодом, що дає змогу автоматизувати процес розгортання програмного забезпечення та врахувати необхідні інфраструктурні особливості. В загальному випадку, архітектуру Docker показано на рис. 3.12.

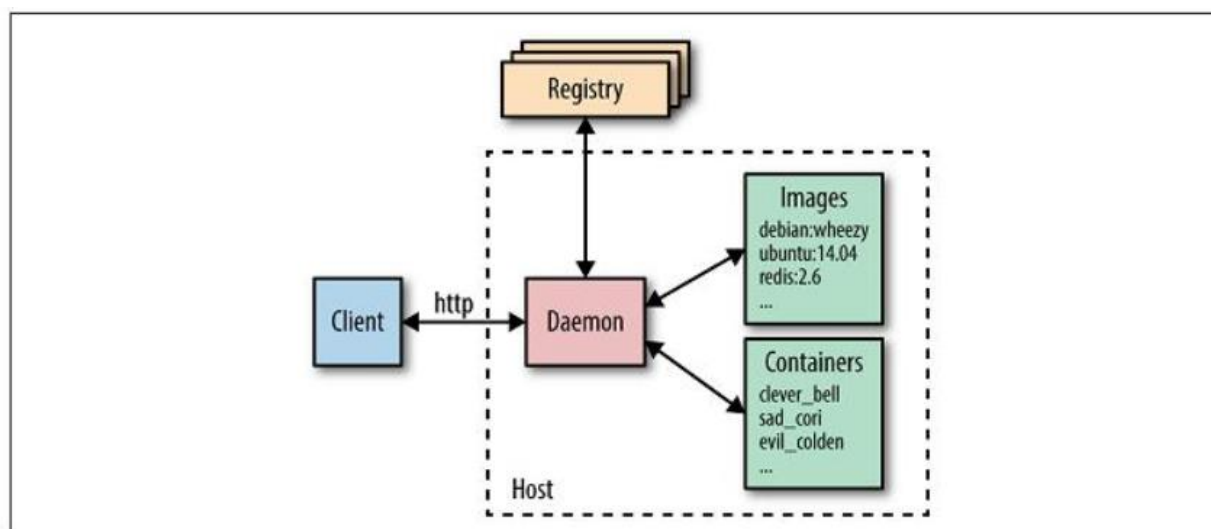


Рис. 3.12. Концептуальна архітектура Docker

Docker відрізняється від віртуальних машин тим, що він не є додатковою операційною системою, встановленою поверх основної системи хоста, і швидкість завантаження в рази більша, ніж в операційних систем. При використанні docker, ядро операційної системи розподіляє ресурси між усіма контейнерами, які представляються у вигляді окремих процесів ОС хоста. На рис. 3.13 наведено архітектурні відмінності між docker та віртуальними машинами.

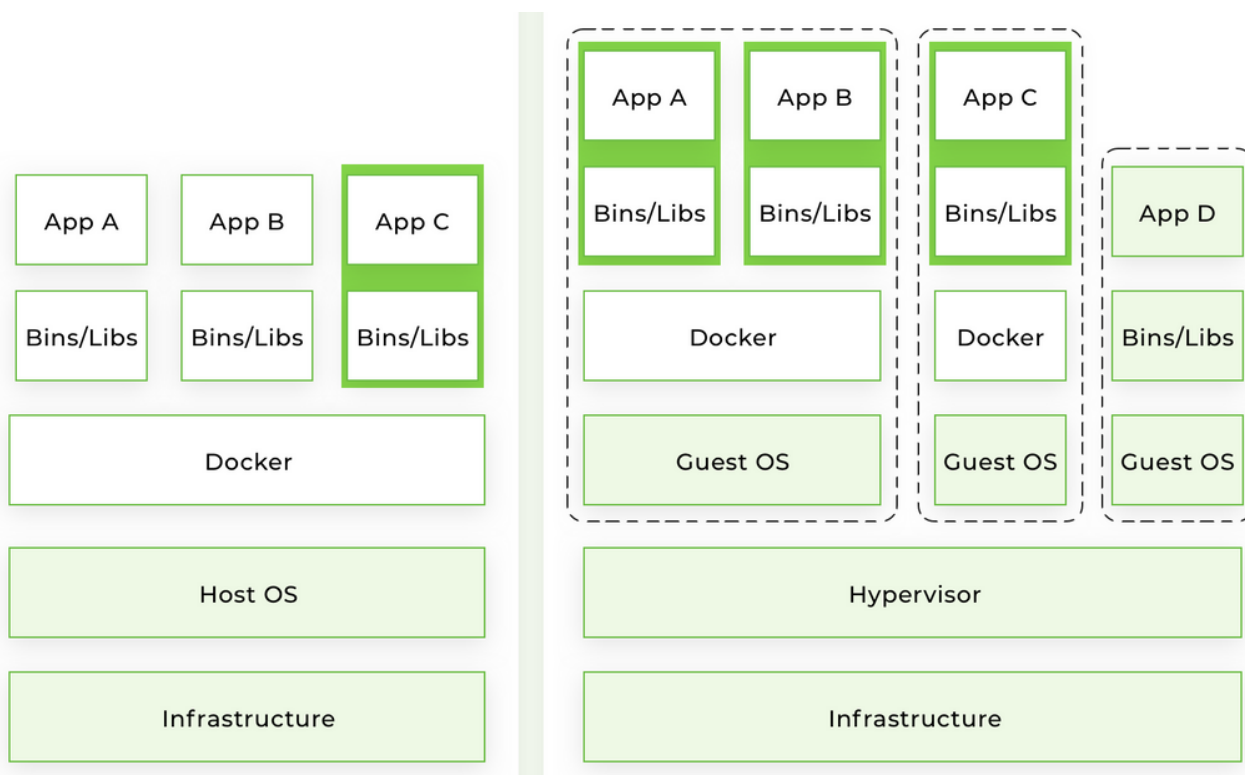


Рис. 3.13. Архітектурні відмінності між Docker та VM

Як видно з рис. 4.8 віртуальні машини є складнішими та повільнішими агентами при розгортанні моделей прогнозування трафіку у комп'ютерних мережах написаними мовою Python.

Docker забезпечує наступні переваги:

- прискорення процесу імплементації програмного забезпечення;
- забезпечення зручності інкапсуляції програмних систем і додатків;
- забезпечення однотипної поведінки програмного продукту у різних середовищах;

- простота управління та моніторингу за інфраструктурою та самим додатком;
- зручність інструментальних засобів при масштабуванні.

Щодо прискорення процесу імплементації, то docker дозволяє запускати необхідні бібліотеки і середовища розробки з контейнера. Крім того, він забезпечує можливість одночасного запуску різних версій одного і того ж ПЗ.

Зручність інкапсуляції у Docker забезпечується шляхом формування єдиного образу для розгортання у різних операційних системах та сервісах.

Однакова поведінка одного і того ж програмного забезпечення запованого у docker на різних ОС реалізується шляхом побудови і контролю залежностей інфраструктури, що зводить до мінімуму ймовірність появи помилок.

Контроль та управління вмістом docker контейнерів забезпечується наявністю системи логування, що дає змогу централізовано збирати інформацію про стан і звернення додатків. Окрім того, можна інтегрувати сторонній зовнішній драйвер, що надасть можливість зберігати та контролювати логи у заданому місці.

Масштабування, у випадку правильного проектування програмного забезпечення та поміщення у контейнер, виконується шляхом використання принципів конфігурації у різних середовища, передачі пакетів на основі TCP/UDP.

Оскільки, операційна система Linux є «рідною» для Docker, то він ефективно може використовувати властивості ядра, а для інших ОС – можна запускати у вигляді не великої віртуальної машини. Це стосується систем macOS і Windows.

Для запуску Docker на базі платформи Linux, необхідно виконувати всі команди з правами root або приєднати поточного користувача до групи docker. На рис. 3.14 наведено запуск Docker.

```
sudo usermod -aG docker $(whoami)
```

Рис. 3.14. Запуск Docker в ОС Linux

Для того, щоб сформувати образ Docker необхідно створити файл під назвою Dockerfile, що представляє собою текстовий файл з набором інструкцій.

Образ, для одержання вмісту з веб-ресурсу і зберігання його у текстовому файлі показано на рис. 3.15.

```
FROM ubuntu:latest
RUN apt-get update
    && apt-get install --no-install-recommends --no-install-suggests -y curl
    && rm -rf /var/lib/apt/lists/*
ENV SITE_URL http://example.com/
WORKDIR /data
VOLUME /data
CMD sh -c "curl -Lk $SITE_URL > /data/results"
```

Рис. 3.15. Образ Docker

У результаті виконання коду одержують Dockerfile і після цього можна перейти до формування збірки образу. Для того, щоб створити збірку необхідно перейти у папку examples/curl і запустити на виконання команду, яка наведена на рис. 3.16.

```
docker build . -t test-curl
```

Рис. 3.16. Приклад створення збірки

Результат формування образу збірки представлено на рис. 3.17.

```

Sending build context to Docker daemon 3.584kB
Step 1/6 : FROM ubuntu:latest
--> 113a43faa138
Step 2/6 : RUN apt-get update && apt-get install --no-install-recommends --no-install-sugg
--> Running in ccc047efe3c7
Get:1 http://archive.ubuntu.com/ubuntu bionic InRelease [242 kB]
Get:2 http://security.ubuntu.com/ubuntu bionic-security InRelease [83.2 kB]
...
Removing intermediate container ccc047efe3c7
--> 8d10d8dd4e2d
Step 3/6 : ENV SITE_URL http://example.com/
--> Running in 7688364ef33f
Removing intermediate container 7688364ef33f
--> c71f04bdf39d
Step 4/6 : WORKDIR /data
Removing intermediate container 96b1b6817779
--> 1ee38cca19a5
Step 5/6 : VOLUME /data
--> Running in ce2c3f68dbbb
Removing intermediate container ce2c3f68dbbb
--> f499e78756be
Step 6/6 : CMD sh -c "curl -Lk $SITE_URL > /data/results"
--> Running in 834589c1ac03
Removing intermediate container 834589c1ac03
--> 4b79e12b5c1d
Successfully built 4b79e12b5c1d
Successfully tagged test-curl:latest

```

Рис. 3.17. Результат формування образу збірки

Для з'єднання і забезпечення комунікації між контейнерами можна скористатися CLI-утилітою Docker compose. Для цього використовується рір (рис. 3.18), а композиція Python і Redis наведена на рис. 3.19.

```

sudo pip install docker-compose

```

Рис. 3.18. Інсталяція Docker compose

```

version: '3.6'
services:
  app:
    build:
      context: ./app
    depends_on:
      - redis
    environment:
      - REDIS_HOST=redis
    ports:
      - "5000:5000"
  redis:
    image: redis:3.2-alpine
    volumes:
      - redis_data:/data
volumes:
  redis_data:

```

Рис. 3.19. Композиція Python і Redis

Для запуску взаємодії між Python і Redis використовується команда, яка показана на рис. 3.20.

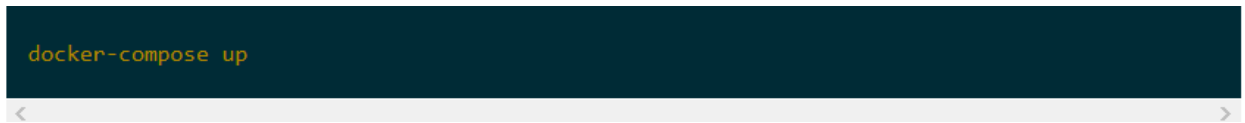
A screenshot of a terminal window with a dark background. The text 'docker-compose up' is displayed in a light-colored font. The terminal has a scrollbar on the right side.

Рис. 3.20. Запуск взаємодії Python і Redis

У результаті покрокового виконання наведених вище команд можна одержати контейнер з реалізованою у Python моделлю, що дозволить запустити її у будь-якому середовищі і її поведінка буде прогнозованою.

3.4. Висновки до розділу

1. Проведено аналіз предметної щодо реалізації програмного засобу прогнозування трафіку у комп'ютерних мережах, у результаті якого визначено множину функціональних вимог, що дало змогу в подальшому побудувати його архітектуру та забезпечити можливість керування параметрами глибокої нейронної мережі.

2. Спроектовано архітектуру та імплементовано засобами мови програмування Python і бібліотек машинного навчання моделі прогнозування трафіку комп'ютерних мереж, які відрізняються кількістю нейронів у прихованих шарах, що дало змогу експериментально обґрунтувати саме архітектуру DBN для короткострокового прогнозування трафіку за робастним алгоритмом і точністю на рівні 97-99%.

3. На основі технології контейнеризації та інструментів Docker створено контейнер з необхідною інфраструктурою для використання реалізованої моделі нейронної мережі сторонніми програмними сервісами, що дало змогу забезпечити її кросплатформність і гнучкість застосування.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Охорона праці

Мета дипломної роботи магістра полягає у дослідженні методів і засобів прогнозування трафіку у комп'ютерних мережах. При роботі з системою прогнозування трафіку потрібно забезпечити дотримання вимог з охорони праці, техніки безпеки та протипожежної безпеки при використанні ПК.

Основними регламентуючими нормативними документами охорони праці користувачів комп'ютерів є:

- НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями»;
- ДСанПіН 3.3-2.007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин»;
- НАПБ А.01.001-2004 «Правила пожежної безпеки в Україні».

Вимоги до приміщень, згідно з [26, 27], щодо розташування робочого місця передбачають виконання наступних вимог:

- мінімальна площа, яка виділяється на одне робоче місце повинна становити мінімум $6,0 \text{ м}^2$, при об'ємі – мінімум $20,0 \text{ м}^3$;
- розташування робочих місць користувачів ПК заборонено у цокольних або підвальних приміщеннях.

При організації робочих місць у НПАОП 0.00-7.15-18 передбачено наявність природного і штучного освітлення. Зазвичай, природне освітлення поступає у приміщення через вікна та світлові прорізи і забезпечує коефіцієнт освітленості на рівні не менше 1,5%. Орієнтація вікон – на північ або північний схід. Штучне освітлення забезпечують відповідні джерела, наприклад, люмінесцентні лампи.

Приміщення з комп'ютерною технікою не повинні межувати з будівлями, де рівень шуму чи вібрації перевищує визначені допустимі значення. Покриття підлоги повинне бути матовим з коефіцієнтом відбиття 0,3-0,5. Для внутрішнього

оздоблення приміщень слід використовувати дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі 0,7-0,8, для стін 0,5-0,6 [26].

У приміщеннях, де організовано робочі місця користувачів ПК, повинні бути забезпечені аптечками першої медичної допомоги. Вологе прибирання у таких приміщеннях є обов'язковим кожного дня.

Щодо ергономічної організації робочого місця, то воно також повинно відповідати вимогам, наведеним у [23, 24]. Конструкція робочого місця повинна забезпечити підтримання оптимальної робочої пози. У відповідності до НПАОП 0.00-7.15-18, обладнання і організація робочого місця працюючих з ЕОМ мають забезпечувати відповідність конструкції всіх елементів робочого місця та їх взаємного, розташування ергономічним вимогам з урахуванням характеру і особливостей трудової діяльності.

Висота робочого столу з ПК повинна бути виконана в діапазоні 680...800 мм, а ширина і глибина – 600...1400 мм і 800..1000 мм відповідно. Стіл також повинен мати достатній простір для ніг, що забезпечить зручну осанку користувача.

Стілець на робочому місці користувача ПК повинен бути підйомно-поворотним, регульованим за висотою, за кутом і за нахилом сидіння та спинки [26].

Екран комп'ютера повинен бути розміщений на відстані 600...700 мм від очей користувача. Розташування монітору має забезпечувати зручність зорового спостереження у вертикальній площині під кутом +30 градусів до нормальної лінії погляду працівника [26].

Електромережі штепсельних з'єднань та електророзеток для живлення ПК потрібно виконувати за магістральною схемою. При організації робочих місць електромережу штепсельних розеток для живлення ПК у центрі приміщення прокладають у каналах або під знімною підлогою в металевих трубах або гнучких металевих рукавах [26].

Щодо безпеки при роботі з ПК, щодня перед початком роботи необхідно очищати монітор від пилу та інших забруднень. Після закінчення роботи з ПК, він та периферійні пристрої повинні бути відключені від електричної мережі. У разі

виникнення певної аварійної ситуації необхідно негайно відключити ПК від електричної мережі. Не допускається виконувати обслуговування, ремонт та налагодження ПК безпосередньо на робочому місці [26].

Основні вимоги до пожежної безпеки вказані в НАПБ А.01.001-2004 «Правила пожежної безпеки в Україні». Згідно з [26], на та під приміщеннями, в яких розміщені ЕОМ, а також у суміжних із ними приміщеннях не дозволяється розташування приміщень категорій А та Б за вибухопожежною небезпекою.

Фальшпідлога у приміщеннях з ЕОМ має бути з негорючих матеріалів або матеріалів груп горючості Г1, Г2 з межею вогнестійкості не менше 0,5 години. Простір під нею слід розділяти негорючими діафрагмами на відсіки площею не більше 250 м². Діафрагми повинні мати межу вогнестійкості не менше 0,75 год. Звукопоглинаюче облицювання стін та стель цих приміщень слід виготовляти з негорючих матеріалів або матеріалів груп горючості Г1, Г2.

Персональні комп'ютери після закінчення роботи повинні відключатися від мережі. Не рідше одного разу на квартал необхідно очищати від пилу агрегати та вузли, кабельні канали та простір між підлогами [27].

Приміщення повинні бути забезпечені первинними засобами пожежогашіння, а саме вогнегасниками, що використовуються для локалізації і ліквідації пожеж у їх початковій стадії розвитку.

Вогнегасники слід встановлювати у легкодоступних та помітних місцях (коридорах, біля входів або виходів з приміщень тощо), а також у пожежонебезпечних місцях, де найбільш вірогідна поява осередків пожежі. При цьому необхідно забезпечити їх захист від попадання прямих сонячних променів та безпосередньої (без загороджувальних щитків) дії опалювальних та нагрівальних приладів.

Вибір типу та необхідна кількість вогнегасників визначається відповідно до Типових норм належності вогнегасників, затверджених наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 02.04.2004 № 151.

У дипломній роботі досліджено методи і засоби прогнозування трафіку у комп'ютерних мережах, тому важливим та актуальним було провести аналіз основних вимог до приміщень та робочих місць з ПК, що дозволило забезпечити комфортні і безпечні умови праці інженерів комп'ютерних систем. Також проаналізовано правила електробезпеки під час роботи з ПК та вимоги до пожежної безпеки в приміщенні.

4.2. Шум, вібрація, ультразвук, електромагнітні випромінювання у виробничих приміщеннях для роботи з ВДТ та захист від них

Під шумом розуміють набір багаточисельних звуків, які швидко змінюються за частотою, силою і складаються з ряду гармонік [28]. З фізичної точки зору звуки є механічними коливальними рухами частинок пружного середовища в діапазоні частот, що чує людина. Звукові гармоніки розповсюджуються у вигляді хвиль. Шум є загально-біологічним подразником, діє не тільки на органи слуху, але може викликати порушення роботи серцево-судинної і нервової систем, зумовлювати професійні захворювання [28]. Основними характеристиками звукових коливань є інтенсивність (сила), частота і форма звукової хвилі. Інтенсивність визначається енергією, що переноситься за 1 с звуковою хвилею через поверхню площею 1 м^2 , яка перпендикулярна напрямку розповсюдження звукової хвилі. Діапазон тисків, що сприймає вухо людини, дуже широкий ($10\text{-}12 \text{ Вт/м}^2$ – поріг больового відчуття, верхня межа) [28].

З розвитком промисловості все більший контингент людей підпадає під вплив вібрацій, які являють собою механічні коливання, що передаються тілу людини. Основні параметри вібрацій – частота та амплітуда коливань, але на відміну від шуму, при якому енергія механічних коливань передається через повітряне середовище, при дії вібрацій вона розповсюджується по тканинах і викликає їх коливання або тіла людини в цілому [28]. Найбільш небезпечна вібрація частотою 16-250 Гц, дія якої призводить до вібраційної хвороби. Нормування шуму здійснюється згідно з “Санітарними нормами допустимих рівнів

шуму на робочих місцях”. В Україні застосовується принцип нормування шуму на основі граничних спектрів (гранично допустимих рівнів звукового тиску) в октавних смугах частот та еквівалентних рівнів звуку. Гранично-допустимі рівні шумів санітарними нормами встановлені для кожного класу [28]:

- для високочастотних шумів (вище 800 Гц) – 75-85 дБ;
- для середньо частотних шумів (300-800 Гц) – 85-90 дБ;
- для низькочастотних шумів (до 300 Гц) – 90-100 дБ.

Шумові явища мають якість кумуляції, накопичуючись в організмі, вони все більше і більше пригнічують нервову систему. Відомо, що після шумової дії інтенсивністю 120 дБ протягом однієї години потрібно 5 годин, щоб гострота слуху повернулась до норми. Стабільні широкосмугові шуми, які перевищують граничний рівень, викликають зниження темпу, ефективності й якості роботи операторів [28].

Ультразвук широко застосовують у технологічних процесах виготовлення радіоелектронної апаратури (промивка деталей, зварювання мініатюрних вузлів тощо) з частотою вище 2220 кГц. При цьому густина енергії ультразвукових коливань у мільйони разів більша густини енергії звуків, які ми чуємо. Тому під його дією відбувається нагрівання тіла, а при дії коливань через рідкі і тверді середовища відбувається розривання і руйнування тканин [28].

Захист від ультразвуку, який діє через повітряне середовище, досягається шляхом звукоізоляції установок (листова сталь, дюралюміній, що обклеєні гумою або руберойдом, гетинакс) або розміщення їх в окремій звукоізолюючій кабіні. Ультразвукові установки повинні мати блокування, яке відключає генератор ультразвукових коливань в момент відкривання кришок або кожухів [28].

Для запобігання шкідливої дії шуму і вібрації на організм працюючих проводяться технічні, організаційні і медико-профілактичні заходи. Одним з основних технічних заходів є зменшення при експлуатації та на стадії проектування, конструювання обладнання причин шуму і вібрації в самому джерелі утворення. Досягають цього завдяки використанню раціональної конструкції обладнання, заміни ударної дії деталей і машин коливальною,

з'єднання елементів гнучкими зв'язками, врівноважування обертових частин механізмів, заміни металевих деталей пластмасовими, забезпечення різних власних частот коливань механізму з частотою збуджуючої сили [28].

Якщо неможливо ізолювати чи знизити шум і вібрацію самого джерела, потрібно:

- ізолювати джерело шуму або вібрації від навколишнього середовища засобами вібро- та звукоізоляції;
- раціонально планувати виробничі приміщення, що мають інтенсивні джерела шуму;
- збільшувати звукопоглинання внутрішніх поверхонь приміщення шляхом звукопоглинальних покриттів.

Якщо не вдається зменшити рівень шуму і вібрації на робочому місці до нормативних значень та необхідно використовувати засоби індивідуального захисту: рукавиці, взуття, навушники, м'які шоломи, які зменшують рівень звукового тиску на 40-50 дБ.

У процесі виробництв, експлуатації і зберігання комп'ютерної і радіоелектронної апаратури можуть виникати механічні і динамічні дії, що характеризуються широким діапазоном частот коливань, а також амплітудою, прискоренням і часом дії [28].

При експлуатації високочастотного обладнання всередині виробничих приміщень зниження напруженості електромагнітного випромінювання досягається такими методами:

- захист часом – обмеження часу перебування людини в електромагнітному полі, що залежить від інтенсивності опромінення або напруженості ЕМП.
- захист відстанню застосовується при неможливості послабити інтенсивність опромінення в заданій зоні іншими методами: збільшують відстань між джерелом випромінювання і обслуговуючим персоналом;
- добре виконане екранування джерела і усунення нещільності у фланцевих з'єднаннях, фідерів, зазорів у обшивці корпусів, нещільних електричних контактів;

- проведення дистанційного контролю й управління роботою передавачів з екранованого приміщення;
- засобами індивідуального захисту.

В залежності від типу джерела випромінювання, його потужності, характеру технологічного процесу може застосовуватись один з вказаних методів або будь-яка їх комбінація.

ВИСНОВКИ

Основні наукові та практичні результати полягають в наступному.

1. Проведено аналіз принципів організації та особливостей формування трафіку у комп'ютерних мережах, що дало змогу визначити фактори і можливі методи класифікації потоків даних, які циркулюють у мережі. Встановлено, що найбільш ефективним способом визначення типу трафіку є статистичні моделі класифікації з учителем, які в подальшому можна використати при прогнозуванні трафіку.

2. Проаналізовано сучасні підходи, моделі і методи прогнозування трафіку у комп'ютерних мережах і встановлено, що в залежності від терміну прогнозування та кількості наявних історичних даних, найбільш ефективними є регресійні моделі, моделі і методи аналізу часових рядів, нейромережеві підходи.

3. Проведено аналіз функціоналів якості при прогнозуванні трафіку у комп'ютерних мережах і запропоновано використовувати не один, а декілька показників якості, що дає змогу оцінити точність алгоритму прогнозування, його робастність (стійкість результатів прогнозування) та інші.

4. Проаналізовано підхід для прогнозування часових рядів на основі методів екстраполяції, що дало змогу обґрунтувати ефективність їхнього застосування, зокрема методу ковзного середнього, для побудови прогнозів у випадку, коли трафік у мережі є стабільним та регулярним у часі.

5. Досліджено методи регресійного аналізу в контексті прогнозування трафіку у комп'ютерній мережі, у результаті якого встановлено, що він є найбільш використовуваним на практиці і базується на визначенні коефіцієнтів регресії та оцінюванні точності прогнозу, при цьому припускають, що коефіцієнти регресії є сталими.

6. Запропоновано застосування моделей ARMA та APIMA для прогнозування часових рядів при аналізі трафіку локальних комп'ютерних мереж, коли трафік передачі та приймання пакетів є однорідним і взаємодія між вузлами мережі приблизно рівномірна, що дало змогу забезпечити високу точність

прогнозування трафіку у коротко і середньо- строковому інтервалах часу на рівні $\geq 95\%$ без врахування сезонної і трендової складової.

7. Запропоновано та спроектовано архітектури глибоких нейронних мереж, вхідний шар яких містить 7 нейронів, кількість проміжних шарів становить 4 рівні, вихідний шар містить один нейрон, що дало змогу практично в реальному часі забезпечити високу точність прогнозування (97%) трафіку для інтернет-сервіс провайдерів на інтервал 1 год вперед.

8. Проведено аналіз предметної щодо реалізації програмного засобу прогнозування трафіку у комп'ютерних мережах, у результаті якого визначено множину функціональних вимог, що дало змогу в подальшому побудувати його архітектуру та забезпечити можливість керування параметрами глибокої нейронної мережі.

9. Спроектовано архітектуру та імплементовано засобами мови програмування Python і бібліотек машинного навчання моделі прогнозування трафіку комп'ютерних мереж, які відрізняються кількістю нейронів у прихованих шарах, що дало змогу експериментально обґрунтувати саме архітектуру DBN для короткострокового прогнозування трафіку за робастним алгоритмом і точністю на рівні 97-99%.

10. На основі технології контейнеризації та інструментів Docker створено контейнер з необхідною інфраструктурою для використання реалізованої моделі нейронної мережі сторонніми програмними сервісами, що дало змогу забезпечити її кросплатформність і гнучкість застосування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Feng, H., Shu, Y.: Study on network traffic prediction techniques. In: International Conference on Wireless Communications, Networking and Mobile Computing, vol. 2. 2005. pp. 1041–1044.
2. Park, K., Willinger, W. (eds.): Self-similar Network Traffic and Performance Evaluation, pp. 94–95. Wiley, New York (2000)
3. Floyd, S., Paxson, V.: Difficulties in simulating the internet. *IEEE/ACM Trans. Netw. (TON)* 9(4). 2001. pp. 392–403.
3. Shang, P., Li, X., Kamae, S.: Nonlinear analysis of traffic time series at different temporal scales. *Phys. Lett. A* 357(4), 2006. pp. 314–318.
4. Осухівська Г., Лобур Т., Білостоцький Т. Дослідження та моделювання інтернет-трафіку комп'ютерної мережі. Збірник тез доповідей XVI наукової конференції Тернопільського національного технічного університету імені Івана Пулюя. 2012. С. 58.
5. Білостоцький Т., Осухівська Г. Математичне моделювання передачі даних в комп'ютерних мережах. Матеріали II науково-технічної конференції „Інформаційні моделі, системи та технології“. 2012. С. 36.
6. Romeu, P., Zamora-Martínez, F., Botella-Rocamora, P., Pardo, J.: Time-series forecasting of indoor temperature using pre-trained deep neural networks. In: International Conference on Artificial Neural Networks. Springer, Heidelberg. 2013. pp. 451–458.
7. Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H. Greedy layer-wise training of deep networks. *Adv. Neural. Inf. Process. Syst.* 19. 2007. 153 p.
8. X.-W. Chen and X. Lin. Big Data Deep Learning: Challenges and Perspectives. *IEEE Access*, Vol.2. 2014. pp. 514–525.
9. Shirchorshidi A. S. Big data clustering: a review. International Conference on Computational Science and Its Applications. Springer, Cham, 2014. pp. 707-720.
10. Kurasova O. Strategies for big data clustering/ O. Kurasova et al. // 2014 IEEE 26th International Conference on Tools with Artificial Intelligence. IEEE, 2014. pp. 740-747.

11. Lin J., Kolcz A. Large-scale machine learning at twitter. Proc. ACM SIGMOD Scottsdale Arizona USA. 2012. pp. 793-804.
12. B. Panda et al. MapReduce and its application to massively parallel learning of decision tree ensembles // *Scaling Up Machine Learning: Parallel and Distributed Approaches*. 2012. P.
13. Krizhevsky A et al. ImageNet classification with deep convolutional neural networks // In Proc. Advances in Neural Information Processing Systems. 2012. 25. pp. 1090–1098.
14. Hinton G. et al. Deep neural network for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*. 2012. № 29. pp. 82–97.
15. Li Deng and Dong Yu. Deep Learning: Methods and Applications. *Foundations and Trends in Signal Processing*, Vol. 7. N. 3-4. 2014. pp. 197–387.
16. Schmidhuber J. Deep learning in neural networks: An overview. *Neural Networks*. Vol. 61. N. 1. January 2015. pp. 85–117.
17. Fischer and C. Igel. Training Restricted Boltzmann Machines: An Introduction. *Pattern Recognition*. Vol. 47, N. 1. January 2014. pp. 25–39,
18. Bengio Y. Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35. N. 8. August 2013. pp. 1798–1828.
19. LeCun, Y., Bengio, Y., Hinton, G. Deep learning. *Nature*. 2015. 521 (7553). pp. 436–444.
20. Golovko V. The Nature of Unsupervised Learning in Deep Neural Networks: A New Understanding and Novel Approach. *Optical Memory and Neural Networks (Springer Link)*. 2016. Vol. 25. № 3. pp. 127–141.
21. Golovko, V.A. Deep learning: an overview and main paradigms. *Optical Memory and Neural Networks (Information Optics)*. Vol. 26, № 1. 2017. pp. 1–17.
22. Liu, J.N., Hu, Y., You, J.J., Chan, P.W.: Deep neural network based feature representation for weather forecasting. In: *International Conference on Artificial Intelligence (ICAI)*. 2014. p. 10.

23. Dalto, M.: Deep neural networks for time series prediction with applications in ultra-short-term wind forecasting. In: IEEE ICIT.2015. pp. 257-268.
24. Oliveira, T.P., Barbar, J.S., Soares, A.S.: Multilayer Perceptron and Stacked Autoencoder for Internet Traffic Prediction. In: IFIP International Conference on Network and Parallel Computing. Springer, Heidelberg. 2014. pp. 61–71.
25. Huang, W., Hong, H., Li, M., Hu, W., Song, G., Xie, K.: Deep architecture for traffic flow prediction. In: International Conference on Advanced Data Mining and Applications. Springer, Heidelberg. 2013. pp. 165– 176.
26. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». Київ. 2018.
27. Катренко Л.А., Катренко А.В. Охорона праці в галузі комп'ютерингу. Львів: Магнолія-2006. 2012. 544 с.
28. Желібо Е.Н. Безпека життєдіяльності: Навчальний посібник. Київ: «Каравела», Львів: «Новий світ - 2000». 2001. 320с.

Додаток А

Текст наукових публікацій дипломної роботи магістра

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
 Тернопільський національний технічний університет імені Івана Пулюя (Україна)
 Національна академія наук України
 Університет імені П'єра і Марії Кюрі (Франція)
 Маріборський університет (Словенія)
 Технічний університет у Кошице (Словаччина)
 Вільнюський технічний університет ім. Гедимінаса (Литва)
 Шауляйська державна колегія (Литва)
 Жешувський політехнічний університет ім. Лукасевича (Польща)
 Білоруський національний технічний університет (Республіка Білорусь)
 Міжнародний університет цивільної авіації (Марокко)
 Національний університет біоресурсів і природокористування України (Україна)
 Наукове товариство ім. Шевченка
 ГО «Асоціація випускників Тернопільського національного технічного університету імені Івана Пулюя»

АКТУАЛЬНІ ЗАДАЧІ СУЧАСНИХ ТЕХНОЛОГІЙ

Збірник

тез доповідей

Том II

**IX Міжнародної науково-технічної
конференції молодих учених та студентів**

25-26 листопада 2020 року



**УКРАЇНА
ТЕРНОПІЛЬ – 2020**

25.	С.А. Лупенко, В. С. Вівчарик ВИКОРИСТАННЯ ВІДДАЛЕНОЇ ІНЖЕНЕРІЇ В ЗАДАЧАХ МОДЕЛЮВАННЯ ТА ОПРАЦЮВАННЯ ЦИКЛІЧНИХ СИГНАЛІВ	38
26.	А.М. Луцків, В.Ю. Бутинець АНАЛІЗ МЕТОДІВ ПРОГНОЗУВАННЯ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ	40
27.	А.М. Луцків, М.В. Ващук МЕРЕЖІ ПЕТРІ ЯК МЕТОД МОДЕЛЮВАННЯ ДИНАМІЧНИХ КОМП'ЮТЕРНИХ СИСТЕМ	41
28.	Л. М. Магула, С. Попович, О. Р. Іванців, М. І. Яворська МОДЕЛЮВАННЯ РОБОТИ ПРИЛАДОВОЇ СИСТЕМИ ДЛЯ ПОВІРКИ ДЕТАЛЕЙ НА НАЯВНІСТЬ КОМПОЗИТНИХ ВКЛЮЧЕНЬ ЗАСОБАМИ МЕРЕЖІ ПЕТРІ	42
29.	В. П. Марценюк, Н. В. Мілян ОГЛЯД МЕТОДІВ ОПТИМІЗАЦІЇ В МАШИННОМУ НАВЧАННІ: ГРАДІЄНТНИЙ СПУСК ТА СТОХАСТИЧНИЙ ГРАДІЄНТНИЙ СПУСК	44
30.	А. Г. Микитишин, О. С. Голотенко, І.Т.Ярема ДОСЛІДЖЕННЯ ТЕПЛІСТІЙКОСТІ ТА УДАРНОЇ В'ЯЗКОСТІ ЕПОКСИДНОЇ СМОЛИ ПРИ ТРИВАЛІЙ ВИТРИМЦІ	46
31.	П. І. Мойсей, І. Ю. Дезів МЕТОД ОБРОБКИ ЗОБРАЖЕННЯ ДЛЯ ВЕРИФІКАЦІЇ ОСОБИ	47
32.	Д.В. Мурза, Ю.О. Круглик, С.В. Марценко МЕТОДИ ТА ЗАСОБИ ОПТИМІЗАЦІЇ РОБОТИ МЕРЕЖ РІЗНОГО ПРИЗНАЧЕННЯ	48
33.	Д.В. Мурза, Ю.О. Круглик, С.В. Марценко ДОСЛІДЖЕННЯ ВПРОВАДЖЕННЯ НОВИХ ПОСЛУГ У МЕРЕЖАХ ОПЕРАТОРІВ ЗВ'ЯЗКУ ТЕХНОЛОГІЇ 5G	49
34.	О.Б.Назаревич, Т.О. Назаревич ВИКОРИСТАННЯ РАДІО-МОДУЛІВ LORA НА ДЛЯ ВІДДАЛЕНОГО КЕРУВАННЯ БЕЗПЛОТНИКОМ	50
35.	Ю.В. Нестор, І.В. Бойко САМОУЗГОДЖЕНИЙ РОЗРАХУНОК ПОТЕНЦІАЛЬНОГО ПРОФІЛЮ AIN/GAN НАНОСТРУКТУР	52
36.	Р.В. Оленюх, Р.Б. Грембач ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ АВТОМАТИЗОВАНОГО КЕРУВАННЯ ПОЛИВОМ	54

УДК 004.942

А.М. Луцків, канд. техн. наук, доц., В.Ю. Бутинець

Тернопільський національний технічний університет імені Івана Пулюя, Україна

АНАЛІЗ МЕТОДІВ ПРОГНОЗУВАННЯ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ

A.M. Lutskiv PhD, Assoc. Prof., V.Yu. Butynets

ANALYSIS OF TRAFFIC FORECASTING METHODS IN COMPUTER NETWORKS

Враховуючи сучасні загрози життю і здоров'ю людей, які пов'язані з поширенням пандемії COVID-19, а також тенденції до глобалізації, важливість забезпечення надійності та стабільності функціонування комп'ютерних мереж зростає в рази. Тому задача віддаленого моніторингу, управління та прогнозування трафіку у комп'ютерних мережах набуває все більшої актуальності. Окрім цього, алгоритми прогнозування кількості переданих та одержаних пакетів даних повинні бути адаптивними і базуватись на оптимальних інтелектуальних алгоритмах, які б давали змогу при потребі розгорнути додаткову інформаційно-технічну інфраструктуру при зростанні навантаження на вузли комп'ютерних мереж, або згорнути її у випадку зниження до деякого визначеного рівня.

Прогнозування у часі трафіку комп'ютерних мереж передбачає застосування часових рядів. У науковій літературі під часовим рядом розуміють послідовність будь-яких значень деякої величини у різні моменти часу.

Випадковим рядом можна вважати кількість переданих чи/та одержаних даних у комп'ютерній мережі або кількість користувачів, які створюють навантаження на вузли мережі деякого об'єкта (поштовий сервер, сервер дистанційного навчання, файловий сервер, маршрутизатор і т.п.). Часовий ряд, як правило, включає в себе вибірку значень навантаження на вузол комп'ютерної мережі у різні моменти часу або обсяг переданих пакетів даних за певний інтервал часу.

Формалізованими методами прогнозування часових рядів є: екстраполяція прогнозу; регресійний аналіз; прогнозування з використання штучних нейронних мереж; прогнозування на базі АРІМА моделей; адаптивні методи прогнозування; прогнозування з використанням гібридних систем.

Один з найпоширеніших і простих методів є екстраполяція при проведенні прогнозування. Для побудови прогнозів за допомогою екстраполяції збирають статистично важливі тенденції зміни тих чи інших кількісних характеристик досліджуваного об'єкта. Для використання екстраполяції потрібно мати інформацію про історію розвитку об'єкта за період, в 2-3 рази більший періоду прогнозування. Цей метод можна, використовувати для короткострокового прогнозування. Також до методів екстраполяції відносяться такі методи: метод найменших квадратів та його різновиди; метод ковзного середнього; метод експоненційного згладжування.

Метод найменших квадратів передбачає застосування мінімізації суми квадратичних відхилень між фактичними і розрахунковими значеннями.

Метод ковзного середнього застосовується, коли в рядах спостерігаються різкі коливання параметрів за періодами. У таких рядах, як правило, слабкий зв'язок параметрів з часом і неможливо побачити та дослідити тенденції до зміни параметрів. Існують багато способів для забезпечення згладжування, суть яких зводиться до заміни фактичних рівнів динамічного ряду розрахунковими, тобто зниженням коливанням у порівнянні з вхідними даними.

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ**

МАТЕРІАЛИ

VIII НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

**«ІНФОРМАЦІЙНІ МОДЕЛІ,
СИСТЕМИ ТА ТЕХНОЛОГІЇ»**



9–10 грудня 2020 року

**ТЕРНОПІЛЬ
2020**

А. Рогачук, Д. Баран, О. Конончук, ДОСЛІДЖЕННЯ БЕТОНУ РІЗНОЇ МІЦНОСТІ НЕРУЙІНІВНИМ МЕТОДАМИ КОНТРОЛЮ	
А. Rohachuk, D. Baran, O. Kononchuk INVESTIGATION OF CONCRETE OF DIFFERENT STRENGTH BY NON- DESTRUCTIVE TESTING METHODS	200
А. Луцків, В. Бутинець МЕТОДИ АНАЛІЗУ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ	201
А. Lutskiv, V. Butynets METHODS OF COMPUTER NETWORK TRAFFIC ANALYSIS	
А. Горпинюк ХАРАКТЕРИСТИКА РИНКУ МІЖНАРОДНИХ ТРАНСПОРТНИХ ПОСЛУГ В УКРАЇНІ	202
А. Horpunyuk CHARACTERISTICS OF THE MARKET OF INTERNATIONAL TRANSPORT SERVICES IN UKRAINE	203
Зміст	203

УДК 004.942

А.М. Луцків, канд. техн. наук, доцент, В.Ю. Бутинець

(Тернопільський національний технічний університет імені Івана Пулюя)

МЕТОДИ АНАЛІЗУ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ

UDC 004.942

A.M. Lutskiv PhD, Assoc. Prof., V.Yu. Butynets

METHODS OF COMPUTER NETWORK TRAFFIC ANALYSIS

Більшість програмних продуктів, які функціонують у комп'ютерній мережі передають дані через вже відомі порти апаратного забезпечення. У такому випадку, суть задачі класифікації пакетів полягає у знаходженні TCP SYN-пакету для встановлення серверної частини клієнт-серверного з'єднання на основі TCP. Після цього, необхідно одержати результат щодо доступності цільового номера порту для конкретного додатку, який надіслав пакет. По такому ж принципу працює класифікація пакетів на основі UDP, однак з'єднання при цьому не встановлюється.

Найбільш вагомою перевагою даного методу є простота реалізації та швидкість виконання операцій щодо класифікації. Однак даний метод має і ряд недоліків. До них належать:

- відсутність у деяких програмних продуктів власних портів, які зареєстровані в IANA (функція управління IP адресним простором);
- здатність програмних додатків використовувати відмінні від визначених в операційній системі портів для виконання певних функцій, наприклад, використання відмінного від порта «80» в Unix-подібних системах для HTTP-сервера;
- виникнення помилок при шифруванні на IP рівні, що може спровокувати плутанину TCP і UDP заголовків.

Інший підхід базується на тому, що для зменшення залежності від портів і одержання достовірних даних про використовуваний протокол, застосовуються методи відновлення стану сеансу та аналізуються дані щодо вмісту кожного окремого пакету.

Класифікація на основі корисного навантаження для peer-to-peer трафіку передбачає аналіз його сигнатур на прикладному рівні. Застосування такої класифікації пакетів дає змогу зменшити кількість помилок першого і другого роду до 5%.

Класифікацію, базовану на корисному навантаженні, можна організувати шляхом застосування наступних методів перевірки та опрацювання:

- PBNS (Packet Based No State);
- PBFS (Packet Based Per Flow State);
- MBFS (Message Based Per Flow State);
- MBPS (Message Based Per Protocol State).

Статистичні методи дають змогу аналізувати трафік у комп'ютерних мережах за двома підходами: на основі алгоритмів аналізу поведінки трафіку; статистичні методи аналізу трафіку на мережевому і транспортному рівнях.

Основна ціль алгоритмів аналізу поведінки трафіку ґрунтується на визначенні додатків, які генерують найбільшу частину трафіку у комп'ютерній мережі. У результаті такого аналізу можна виявити, яким чином вузли комунікують між собою і яке програмне забезпечення функціонує на них.