

ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

Кваліфікаційна робота

Моделі та методи прогнозування трафіку в корпоративній мережі

Виконав:
студент гр. СПм-23-4
Коробейніков О.Б.

Керівник:
зав. каф. ЕОМ, д.т.н.,
проф. Коваленко А.А.

Мета роботи та завдання

2

Метою роботи є розробка моделі і методу прогнозування трафіку у корпоративній мережі з використанням апарату мереж Петрі, а також їх практична апробація.

Об'єктом дослідження процес управління трафіком у корпоративних мережах.

Завдання:

- аналіз існуючих методів прогнозування трафіку;
- дослідження теоретичних основ мереж Петрі;
- розробка моделей на основі мереж Петрі для прогнозування трафіку;
- експериментальна перевірка запропонованих моделей і оцінка їх ефективності.

Еталонна модель OSI

3

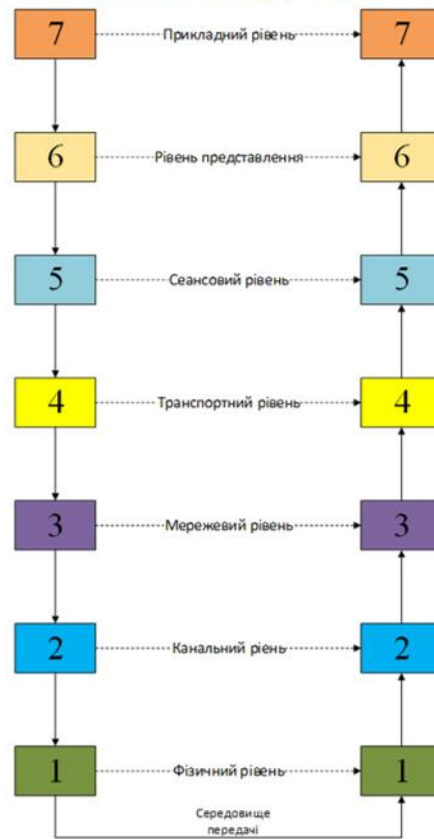
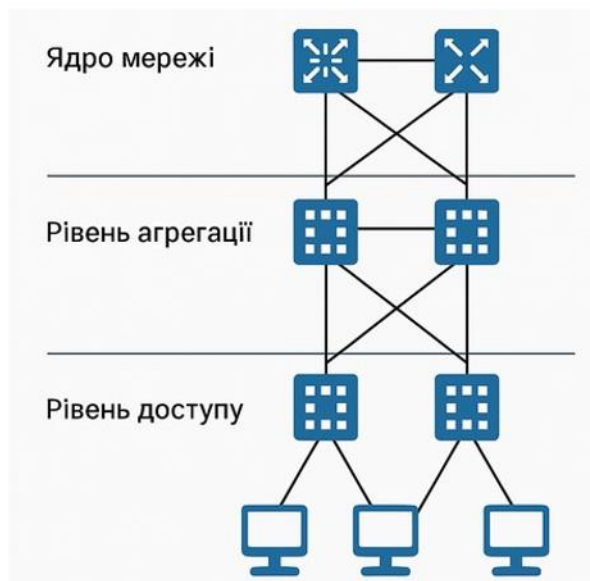


Схема багаторівневої архітектури корпоративної мережі

4



Вимоги до управління трафіком в корпоративній мережі

Моніторинг і аналіз трафіку:

- ✓ виявлення пікових навантажень, відстеження джерел трафіку, ідентифікація «важких» додатків;
- ✓ реалізується за допомогою SNMP, NetFlow, sFlow, DPI (глибокий аналіз пакетів).

Прогнозування навантаження:

- ✓ важливо для планування ресурсів, підвищення ефективності інфраструктури та запобігання перенавантаженню;
- ✓ прогнозування дає змогу адаптувати політики QoS, масштабувати ресурси, змінювати маршрути в залежності від передбачуваного попиту.

Пріоритизація трафіку:

- ✓ забезпечення переваги для критично важливих сервісів (відеоконференції, CRM, VoIP);
- ✓ використання механізмів маркування трафіку (DSCP, CoS), чергування, обмеження пропускної здатності.

Балансування навантаження:

- ✓ розподіл трафіку між кількома серверами чи каналами для зниження затримок і підвищення продуктивності;
- ✓ актуально для систем з великою кількістю запитів до веб-додатків, БД, API.

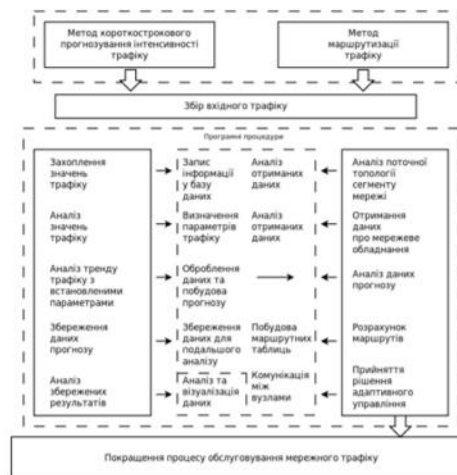
Автоматизоване управління політиками:

- ✓ динамічне регулювання правил доступу, фільтрації, перенаправлення трафіку в залежності від умов;
- ✓ використання SDN (software-defined networking) та мережевої оркестрації.

Інтеграція з інструментами безпеки:

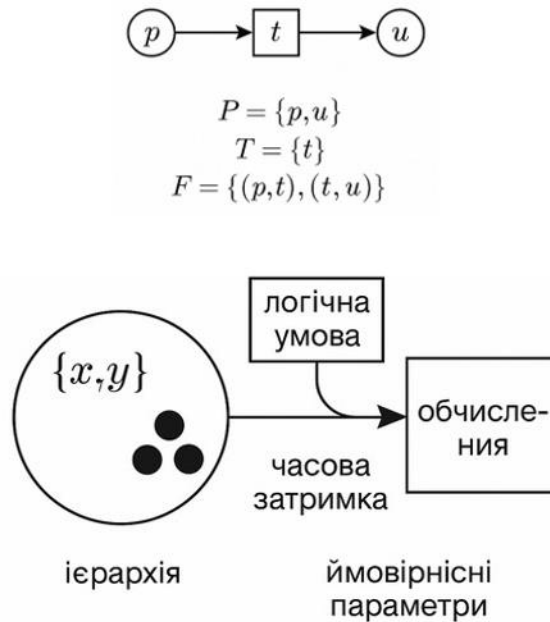
- ✓ контроль SSL/HTTPS-трафіку, аналіз загроз, виявлення підозрілої активності;
- ✓ роль інтегрованих платформ: SIEM, NTA (Network Traffic Analysis), SOAR.

Схема застосування інформаційної технології для завдання прогнозування трафіку. Порівняльний аналіз методів прогнозування



Метод	Точність	Складність реалізації	Швидкість обчислень	Адаптивність
Статистичні моделі	Середня	Низька	Висока	Низька
Нейромережеві методи	Висока	Висока	Низька	Висока
Евристичні підходи	Змінна	Середня	Середня	Висока

Мережа Петрі. Розширена мережа Петрі



Опис моделі та методу прогнозування

Для прогнозування трафіку в корпоративній мережі доцільно використовувати розширену модель мережі Петрі з кольоровими фішками CPN. Вибір цього типу обумовлений необхідністю обробки різних типів трафіку, що мають специфічні характеристики, такі як джерело, призначення, інтенсивність і час передачі. Кожна фішка має внутрішню структуру, що зберігає інформацію про трафік (тип, обсяг, час).

Вузлами мережі є місця, які репрезентують різні сегменти корпоративної мережі (сервери, маршрутизатори, точки доступу) і переходи, які моделюють процес передачі трафіку між вузлами. Стан мережі визначається кількістю та типами фішок у місцях, що відображає інтенсивність трафіку на вузлах.

Дуги зв'язують місця та переходи, визначаючи правила передачі трафіку в системі. Переходи можуть активуватись залежно від умов, які моделюють доступність ресурсів, пропускну здатність каналів, або час передачі.

Прогнозування будується на аналізі історичних даних про переміщення фішок у мережі. Перший крок – це збирання даних про історичний трафік (кількість і тип фішок, інтенсивність передачі). Далі використовуються статистичні або нейромережеві моделі, що аналізують ці дані і визначають закономірності, які дозволяють спрогнозувати майбутні переміщення фішок між місцями.

Зокрема, прогноз може ґрунтуватися на аналізі частоти спрацювання переходів та динаміки змін у маркуванні місць. Отримані результати використовуються для автоматичного налаштування параметрів мережі Петрі, зокрема зміни пропускну здатності каналів або вибору оптимального шляху трафіку.

Метод прогнозування трафіку на основі розширених мереж Петрі

9

Крок 1. Формалізація мережі як моделі Петрі

На цьому етапі створюється базова модель корпоративної мережі у вигляді розширеної мережі Петрі. Формалізація включає: визначення місць – це логічні або фізичні вузли мережі, наприклад сервери, маршрутизатори, точки доступу, сегменти, визначення переходів – це процеси або події, що відбуваються в мережі: передача трафіку, зміна маршруту, фільтрація, балансування, визначення дуг – зв'язків між місцями й переходами, призначення фішок зі структурою (тип трафіку, обсяг, час, джерело/призначення).

Крок 2. Збір історичних даних трафіку

На цьому етапі виконується моніторинг і фіксація фактичних даних про трафік, що проходив через мережу: кількість пакетів у кожному місці; час між активацією переходів; типи переданого трафіку (http, ftp, VoIP); завантаженість вузлів мережі. Дані збираються за певні періоди (наприклад, щогодини або щохвилини) і зберігаються у таблицях.

Крок 3. Аналіз маркування та спрацювань

Далі аналізується, як змінювалось маркування мережі (розподіл фішок) у часі; визначаються частоти спрацювання переходів; вивчаються послідовності переходів; будуються множини станів і сценарії динаміки системи. Це дозволяє виявити, як трафік поширюється по мережі у відповідь на події – наприклад, навантаження на конкретний сегмент призводить до перенаправлення потоків.

Крок 4. Вибір методу прогнозування

На основі зібраних даних обирається алгоритм прогнозування. Є кілька варіантів: статистичні методи (ARIMA, ковзне середнє) – для регулярних трафіків; машинне навчання (регресія, дерева рішень) – для адаптивного прогнозу; нейронні мережі (LSTM) – для складних патернів трафіку.

Крок 5. Побудова прогнозу моделі

Формалізується сам процес прогнозування: вхідні дані: часові ряди трафіку, топологія мережі Петрі; вихідні дані: оцінка інтенсивності трафіку у певних місцях/на дугах; алгоритм: аналіз маркувань, спрацювання переходів, обрахунок майбутніх сценаріїв на основі ймовірностей або функцій.

Прогнозна модель також може інтегрувати часові обмеження, логіку пріоритетів або зважування трафіку за типом.

Крок 6. Реалізація в середовищі моделювання

Модель реалізується в програмному середовищі (Google Colab).

Крок 7. Валідація та коригування

10

Генерація різних типів трафіку

```

np.random.seed(42)
n_points = 1000
time_series = np.arange(1, n_points + 1)

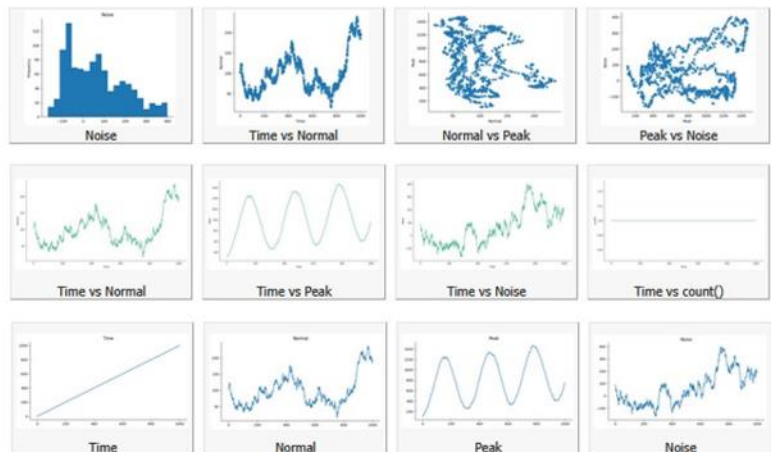
# Типи трафіку: звичайний, піковий, шумний
traffic_normal = 100 + np.cumsum(np.random.normal(0, 1, n_points))
traffic_peak = 100 + np.cumsum(np.random.normal(0, 1, n_points))
traffic_noise = 100 + np.cumsum(np.random.normal(0, 1, n_points))

traffic_df = pd.DataFrame({
    'Time': time_series,
    'Normal': traffic_normal,
    'Peak': traffic_peak,
    'Noise': traffic_noise
})

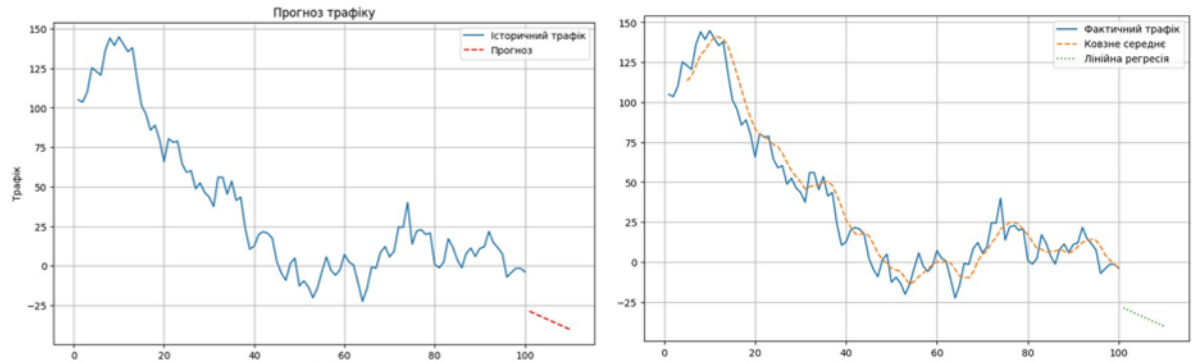
traffic_df.head()

```

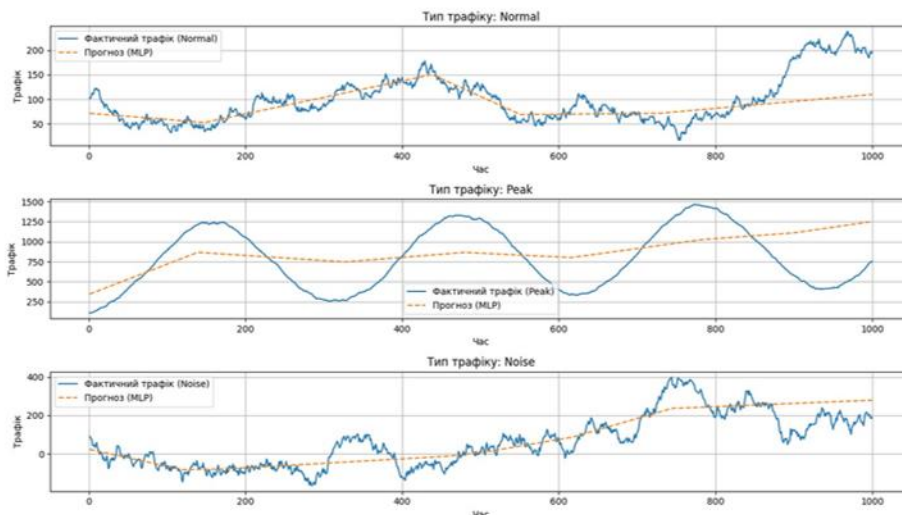
Time	Normal	Peak	Noise	
0	1	102.483571	107.196764	89.872326
1	2	101.792249	112.219826	87.704546
2	3	105.030692	113.117618	75.818247
3	4	112.645841	110.682081	71.198824
4	5	111.475074	115.171531	42.794604



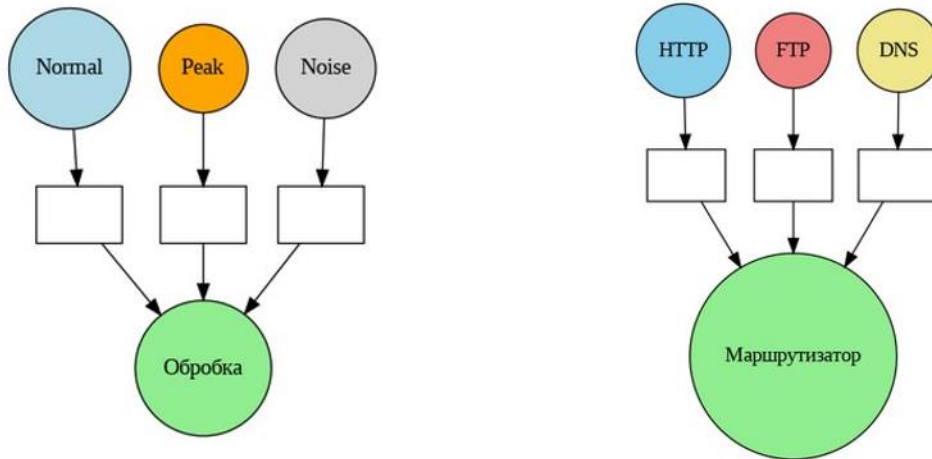
Прогнозування трафіку. Порівняння з ковзним середнім 11



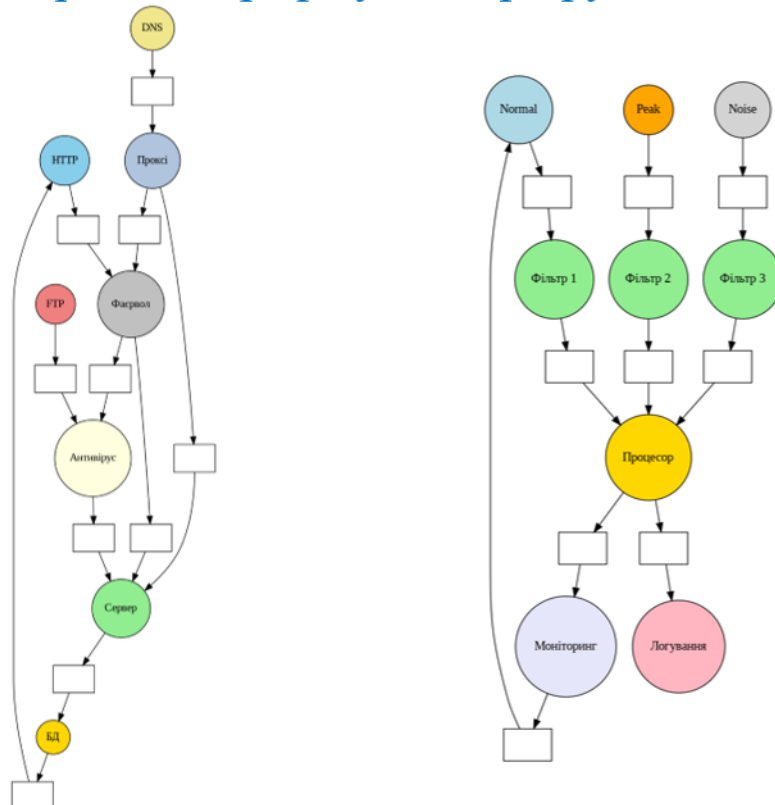
Порівняння реальних і передбачених значень з різними типами трафіку 12



Типи трафіку (Normal, Peak, Noise) та різні типи запитів (HTTP, FTP, DNS) 13



Взаємодія між запитами та сервісами. Обробка різного трафіку та маршрутизація. 14



Апробація результатів

15

DATA PROCESSING METHODS IN A CORPORATE NETWORK

Abstract. Relevance. In the context of modern digital business transformation, corporate networks have become critically important components of enterprise information infrastructure, integrating numerous systems, devices, and services into a unified information environment. Every day, vast volumes of data are generated, transmitted, stored, and processed within these networks, encompassing all aspects of organizational activity – from financial operations and logistics to customer communications and internal management processes. The successful functioning of a corporate network is impossible without effective data processing, which enables the extraction of valuable insights from raw data, supports strategic decision-making, ensures information security, and optimizes resources. Amid intense competition, the rapid growth of information flows, and increasing demands for swift decision-making, data processing methods play a crucial role in enabling the analytical capabilities of enterprises. Data processing methods in corporate environments find applications across diverse business sectors. In the financial domain, they are used for transaction analysis, risk management, market fluctuation forecasting, and fraud detection. In logistics, they support route optimization, warehouse stock control, and real-time supply coordination. In marketing, they facilitate customer behavior analysis, audience segmentation, and personalized communication. In HR departments, they assist in employee performance evaluation and candidate selection. In healthcare, they support patient data analysis, service process optimization, and clinical decision-making. Furthermore, data processing forms the foundation for the development of artificial intelligence systems, digital twins, cybersecurity solutions, and automated enterprise management. Traditional approaches to data processing are gradually being replaced by advanced technologies based on Big Data, real-time stream processing, cloud computing, intelligent data analytics, and machine learning. In this context, issues related to data protection, integrity, availability, and compliance with legal and regulatory requirements for confidentiality gain particular importance. The purpose is to analyze modern methods and technologies for data processing in corporate networks, identify their advantages, disadvantages, and application areas, and classify approaches to data collection, cleaning, storage, and analysis. The subject of research includes methods and technologies for data processing, storage, analysis, and protection. The results. As a result of the conducted research, a comprehensive understanding of modern data processing methods in corporate environments was formed. The architectural models of enterprise information systems were analyzed, key stages of data preparation identified, and approaches to their processing – from classical to intelligent – were characterized. The review revealed a wide range of data processing techniques, from traditional SQL queries and multidimensional analysis to real-time processing and machine learning applications. The choice of a particular approach depends on data volume, data inflow speed, latency sensitivity, and the specific goals of processing.

Keywords: corporate network, data processing, architecture, transaction, cybersecurity, Big Data.

Introduction

The successful operation of modern enterprises largely depends on the efficiency of processing large volumes of information circulating within corporate networks. This involves the collection, systematization, storage, analysis, and transmission of data between various company departments, as well as external counterparts. The complexity of information processing increases with the growing volume of data, the need to ensure prompt access to information, and the tightening requirements for data security and protection. Within a corporate network, data processing encompasses the entire information lifecycle – from its creation or external acquisition to its storage, analysis, and subsequent use in decision-making processes. To fully understand the essence of data processing in this context, it is essential to have a clear understanding of what constitutes a corporate network and the types of data that circulate within it.

A corporate network represents an integrated environment that unites computer systems, servers, storage facilities, network equipment, software, and users into a single information space. Such a network enables uninterrupted communication among all structural units of an organization, regardless of their physical location, allowing for centralized data management, information exchange, and access to resource management, reporting,

and customer support systems.

The data circulating in a corporate environment may vary in format and degree of structure. Structured data includes information arrays with a clearly defined format, such as numerical tables or records in relational databases. Unstructured data, in contrast, lacks a fixed format and may include emails, text documents, audio and video files, images, or social media messages. A separate category is semi-structured data, which contains a certain degree of organization but does not conform to traditional relational database formats.

In this context, data processing entails not only the technical computation of data but also includes stages such as data collection, filtering, aggregation, transformation, analysis, and subsequent presentation in an interpretable format. A key feature of corporate data processing is the necessity to account for complex access hierarchies, user roles, security levels, and adherence to data storage and processing policies in accordance with regulatory requirements.

A defining aspect is that corporate data does not exist in isolation. It is closely integrated into an organization's business processes, forming the context in which it gains significance. This underpins the need for a comprehensive infrastructure capable of handling data processing in alignment with the company's scale, structure, and operational dynamics. Thus, understanding the

entire array of modern enterprise information systems, sensors, web services, or partner company databases.

All these factors create the necessity for a systematic study of processes associated with data handling within corporate networks. Given the growing demands for the speed, reliability, and scalability of information solutions, it is essential to examine not only the general principles of such systems' operation but also the specific methods employed to ensure a complete data cycle – from acquisition to analytical reporting. In this context, the analysis of contemporary approaches to data transformation, selection of processing algorithms, automation tools, and visualization instruments becomes particularly relevant, as they enable not only data storage but also the extraction of practical value for managerial decision-making. The purpose is to analyze modern methods and technologies for data processing in corporate networks, identify their advantages, disadvantages, and application areas, and classify approaches to data collection, cleaning, storage, and analysis.

Main part

The architecture of a corporate data processing system represents a comprehensive, multi-layered structure that ensures interaction between technical, software, and organizational components for the effective management of information flows within an enterprise. Under modern conditions, such an architecture must not only be capable of processing large volumes of data but also of promptly responding to changing business requirements, ensuring scalability, fault tolerance, and uninterrupted access to critical services.

A central element of any corporate architecture is the logic governing the distribution of data and computing resources, which may follow either a centralized or a distributed model. In centralized models, core processing and storage are performed on a single server or within a corporate data center, enabling efficient control over data access, updates, and backup operations. In contrast, distributed systems delegate processing and storage tasks to multiple network nodes, which helps reduce latency, increase flexibility, and bring data closer to end users – particularly in geographically dispersed branches of an enterprise.

The emergence of cloud technologies has led to significant changes in architectural approaches. The virtualization of computing resources, the use of infrastructure as a service and platform as a service [1], have enabled companies to dynamically scale computing power, deploy new services without significant investment in hardware, and integrate with external service providers. In such an environment, data can be stored in cloud repositories, processed through virtual clusters, and analytics can be accessed via web interfaces from anywhere in the world. Despite these advantages, such solutions require special attention to issues of security, access control, data leakage prevention, and compliance with international

Particular attention should be given to the data

An effective architecture for corporate data processing must take into account the specifics of the industry, the scale of the company, the types of data being processed, and the organization's level of digital maturity. Only a flexible, scalable, and secure architecture can enable full automation of analytical processes, reduce infrastructure costs, and significantly improve the quality of managerial decisions based on up-to-date data.

In the context of corporate networks, data processing is impossible without a high-quality and systematic approach to data collection and preparation. This stage determines not only the volume and relevance of available information but also its cleanliness, completeness, and suitability for further analytical processing. In many cases, it is at the collection and preprocessing stages where data suffers the most significant quality losses, which may subsequently lead to erroneous conclusions or ineffective management decisions. Therefore, the question of corporate data collection and preparation methods is of critical importance.

Data in corporate environments can come from a variety of sources, each with its own structure, formatting standards, and update frequency. The main sources include internal enterprise information systems – particularly enterprise resource planning systems [3], accounting and logistics platforms – as well as external web services, API platforms, user behavior monitoring tools, the Internet of Things, and partner databases. Given the high heterogeneity of these sources, the data collection process relies on specialized integration tools that ensure the unification of incoming information and its delivery to a single processing environment.

One of the key methodological approaches to data collection and preparation is the ETL process, which consists of three logically connected stages [4]. The first stage involves extracting data from sources, where it is crucial to ensure data flow continuity, avoid record duplication, and maintain version control. In the transformation stage, data undergoes various operations such as filtering, format standardization, anomaly removal, deduplication, handling of missing values, categorization, or merging of records from multiple tables. The final stage is loading the cleansed and transformed information into target systems, which may include data warehouses, analytical databases, cloud platforms, or local enterprise servers.

Diachenko D., Korobeinikov M., Korobeinikov O., Kovalenko A., Kravchenko P. Data processing methods in a corporate network. Системи управління, навігації та зв'язку, вип.3. Полтава, 2025. С. 81-86.

Висновки

16

Проаналізовані моделі та методи прогнозування трафіку в корпоративній мережі. Проведений аналіз сучасних підходів дозволив встановити, що ефективне управління трафіком вимагає не лише математично коректних моделей, але й адаптивних механізмів, здатних відображати складну динаміку і неоднорідність даних, притаманну реальним мережам. У роботі було реалізовано як базові, так і розширені варіанти мереж Петрі, зокрема – кольорові мережі, що дозволяють урахувати тип трафіку або його джерело як атрибутивні характеристики фішок. Такі моделі виявилися особливо ефективними для моделювання потоків різної природи – нормального, пікового та шумового, що імітує поведінку корпоративної мережі у реальних умовах.

Для прогнозування трафіку було застосовано як класичні методи (ковзне середнє, лінійна регресія), так і підходи на основі машинного навчання, зокрема штучні нейронні мережі. Результати експериментального моделювання засвідчили, що нейромережеві моделі демонструють вищу здатність до узагальнення трендів у складних або зашумлених даних, хоча можуть поступатися простішим алгоритмам у випадках із чітко вираженою періодичністю. Було також підтверджено, що ефективність прогнозу значною мірою залежить від типу вхідного трафіку, що ще раз підкреслює важливість правильної класифікації даних.

Інтеграція моделей прогнозування у структуру мереж Петрі дозволила реалізувати не лише аналітичне, а й симуляційне середовище, придатне для подальших досліджень, зокрема динамічного аналізу навантаження в реальному часі. В роботі представлено приклади візуалізації, реалізовані за допомогою Python та бібліотеки graphviz, що зробило описані процеси більш наочними і прикладними.

ДОДАТОК Б

Програмний код

Б.1 Підготовка середовища та даних

```
# Встановлення бібліотек (за потреби в Colab)
# !pip install matplotlib scikit-learn pandas graphviz
import pandas as pd
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt
import numpy as np
from graphviz import Digraph
from IPython.display import Image
```

Б.2 Генерація різних типів трафіку

```
np.random.seed(42)
n_points = 1000
time_series = np.arange(1, n_points + 1)

# Типи трафіку: звичайний, піковий, шумний
traffic_normal = 100 + np.cumsum(np.random.normal(0, 5,
n_points))
traffic_peak = 100 + np.cumsum(np.random.normal(0, 5, n_points)
+ np.sin(time_series / 50) * 10)
traffic_noise = 100 + np.cumsum(np.random.normal(0, 15,
n_points))

traffic_df = pd.DataFrame({
    'Time': time_series,
    'Normal': traffic_normal,
    'Peak': traffic_peak,
    'Noise': traffic_noise
})

traffic_df.head()
```

Б.3 Прогнозування за допомогою штучної нейронної мережі

```
from sklearn.neural_network import MLPRegressor
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
```

```

X = traffic_df[['Time']]
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

models = {}
predictions = {}

for ttype in ['Normal', 'Peak', 'Noise']:
    y = traffic_df[ttype]
    X_train, X_test, y_train, y_test =
train_test_split(X_scaled, y, test_size=0.1, shuffle=False)

    mlp = MLPRegressor(hidden_layer_sizes=(50, 30),
max_iter=1000, random_state=42)
    mlp.fit(X_train, y_train)

    pred = mlp.predict(X_scaled)
    models[ttype] = mlp
    predictions[ttype] = pred

```

Б.4 Порівняння реальних і передбачених значень

```

plt.figure(figsize=(14, 8))
for i, ttype in enumerate(['Normal', 'Peak', 'Noise']):
    plt.subplot(3, 1, i+1)
    plt.plot(traffic_df['Time'], traffic_df[ttype],
label=f'Фактичний трафік ({ttype})')
    plt.plot(traffic_df['Time'], predictions[ttype],
label='Прогноз (MLP)', linestyle='--')
    plt.title(f'Тип трафіку: {ttype}')
    plt.xlabel('Час')
    plt.ylabel('Трафік')
    plt.legend()
    plt.grid(True)
plt.tight_layout()
plt.show()

```

Б.5 Кольорова мережа Петрі: Типи трафіку (Normal, Peak, Noise)

```

dot = Digraph()
# Місця (типи трафіку)
dot.node('PN', 'Normal', shape='circle', style='filled',
fillcolor='lightblue')
dot.node('PP', 'Peak', shape='circle', style='filled',
fillcolor='orange')
dot.node('PNO', 'Noise', shape='circle', style='filled',
fillcolor='lightgray')

```

```

# Переходи
dot.node('T1', '', shape='box')
dot.node('T2', '', shape='box')
dot.node('T3', '', shape='box')
# Після переходу – обробка
dot.node('PH', 'Обробка', shape='circle', style='filled',
fillcolor='lightgreen')
# Дуги
dot.edge('PN', 'T1')
dot.edge('PP', 'T2')
dot.edge('PNO', 'T3')
dot.edge('T1', 'PH')
dot.edge('T2', 'PH')
dot.edge('T3', 'PH')
dot.render('/mnt/data/colored_petri_traffic', format='png',
cleanup=True)
Image('/mnt/data/colored_petri_traffic.png')

```

Б.6 Кольорова мережа Петрі: Різні типи запитів (HTTP, FTP, DNS)

```

dot = Digraph()
dot.node('P1', 'HTTP', shape='circle', style='filled',
fillcolor='skyblue')
dot.node('P2', 'FTP', shape='circle', style='filled',
fillcolor='lightcoral')
dot.node('P3', 'DNS', shape='circle', style='filled',
fillcolor='khaki')
dot.node('T1', '', shape='box')
dot.node('T2', '', shape='box')
dot.node('T3', '', shape='box')
dot.node('P4', 'Маршрутизатор', shape='circle', style='filled',
fillcolor='lightgreen')
dot.edge('P1', 'T1')
dot.edge('P2', 'T2')
dot.edge('P3', 'T3')
dot.edge('T1', 'P4')
dot.edge('T2', 'P4')
dot.edge('T3', 'P4')
dot.render('/mnt/data/colored_petri_requests', format='png',
cleanup=True)
Image('/mnt/data/colored_petri_requests.png')

```

Б.7 Розширена кольорова мережа Петрі: Обробка різного трафіку та маршрутизація

```

dot = Digraph()
# Місця для типів трафіку

```

```

dot.node('P1', 'Normal', shape='circle', style='filled',
fillcolor='lightblue')
dot.node('P2', 'Peak', shape='circle', style='filled',
fillcolor='orange')
dot.node('P3', 'Noise', shape='circle', style='filled',
fillcolor='lightgray')
# Проміжна маршрутизація
dot.node('P4', 'Фільтр 1', shape='circle', style='filled',
fillcolor='lightgreen')
dot.node('P5', 'Фільтр 2', shape='circle', style='filled',
fillcolor='lightgreen')
dot.node('P6', 'Фільтр 3', shape='circle', style='filled',
fillcolor='lightgreen')
# Обробка
dot.node('P7', 'Процесор', shape='circle', style='filled',
fillcolor='gold')
# Вузли зберігання
dot.node('P8', 'Лопування', shape='circle', style='filled',
fillcolor='lightpink')
dot.node('P9', 'Моніторинг', shape='circle', style='filled',
fillcolor='lavender')
# Переходи
for i in range(1, 10):
    dot.node(f'T{i}', '', shape='box')
# Дуги
edges = [
    ('P1', 'T1'), ('T1', 'P4'),
    ('P2', 'T2'), ('T2', 'P5'),
    ('P3', 'T3'), ('T3', 'P6'),
    ('P4', 'T4'), ('P5', 'T5'), ('P6', 'T6'),
    ('T4', 'P7'), ('T5', 'P7'), ('T6', 'P7'),
    ('P7', 'T7'), ('T7', 'P8'),
    ('P7', 'T8'), ('T8', 'P9'),
    ('P9', 'T9'), ('T9', 'P1') # Кільцева структура для Noise
трафіку
]
for src, dst in edges:
    dot.edge(src, dst)
dot.render('/mnt/data/extended_colored_petri1', format='png',
cleanup=True)
Image('/mnt/data/extended_colored_petri1.png')

```

Б.8 Розширена кольорова мережа Петрі: Взаємодія між запитами та сервісами

```

dot = Digraph()

# Типи запитів
dot.node('P1', 'HTTP', shape='circle', style='filled',
fillcolor='skyblue')

```

```

dot.node('P2', 'FTP', shape='circle', style='filled',
fillcolor='lightcoral')
dot.node('P3', 'DNS', shape='circle', style='filled',
fillcolor='khaki')

# Обробка в сервісах
dot.node('P4', 'Фаєрвол', shape='circle', style='filled',
fillcolor='gray')
dot.node('P5', 'Антивірус', shape='circle', style='filled',
fillcolor='lightyellow')
dot.node('P6', 'Проксі', shape='circle', style='filled',
fillcolor='lightsteelblue')

# Завершення
dot.node('P7', 'Сервер', shape='circle', style='filled',
fillcolor='lightgreen')
dot.node('P8', 'БД', shape='circle', style='filled',
fillcolor='gold')
# Переходи
for i in range(1, 11):
    dot.node(f'T{i}', '', shape='box')
# Дуги
edges = [
    ('P1', 'T1'), ('T1', 'P4'), ('P4', 'T4'), ('T4', 'P7'),
    ('P2', 'T2'), ('T2', 'P5'), ('P5', 'T5'), ('T5', 'P7'),
    ('P3', 'T3'), ('T3', 'P6'), ('P6', 'T6'), ('T6', 'P7'),
    ('P7', 'T7'), ('T7', 'P8'),
    ('P8', 'T8'), ('T8', 'P1'), # Кільцевий маршрут для DNS
    ('P4', 'T9'), ('T9', 'P5'),
    ('P6', 'T10'), ('T10', 'P4') # Міжсервісна маршрутизація
]
for src, dst in edges:
    dot.edge(src, dst)

dot.render('/mnt/data/extended_colored_petri2', format='png',
cleanup=True)
Image('/mnt/data/extended_colored_petri2.png')

```