

"Програмування систем штучного інтелекту"



Analyze
& Learn

Програмування задач аналізу текстів із використанням ШІ

Вступ: Чому аналіз текстів із ШІ важливий?

2.5

Квінтильйони
байтів

Щодня генерується
величезна кількість
текстової
інформації

90%

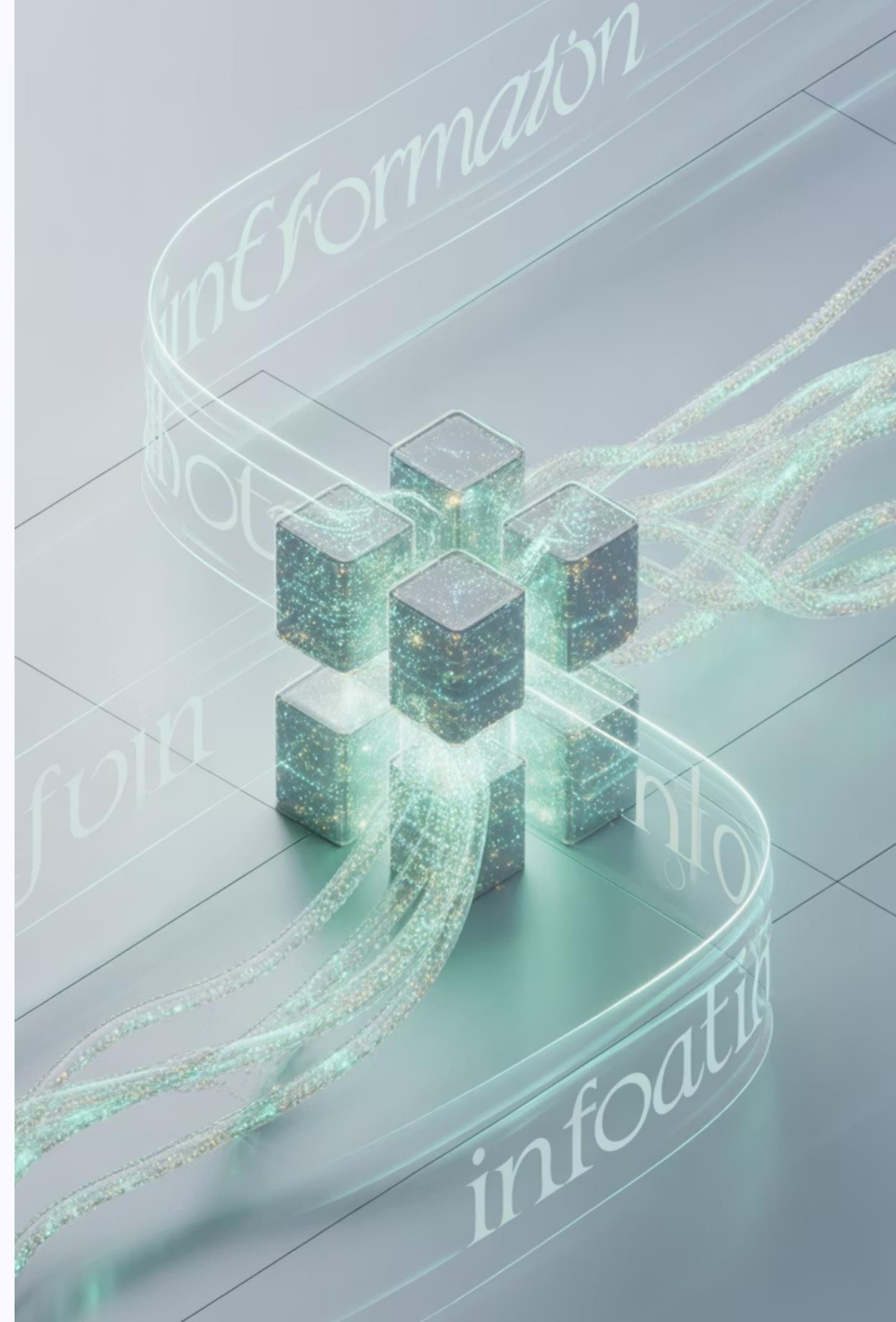
Неструктурованих
даних

Більшість
корпоративної
інформації потребує
обробки ШІ

24/7

Автоматизація
ШІ працює
цілодобово без
втом

ШІ революціонізує обробку тексту: від чат-ботів до автоматичного перекладу та модерації контенту в реальному часі.



Основні задачі аналізу текстів

01 Токенізація та лематизація

Розділення тексту на слова, приведення до базової форми

02 Частотний аналіз

Визначення найважливіших слів та фраз у документі

03 Класифікація текстів

Автоматичне визначення категорії: спам, тональність, тематика

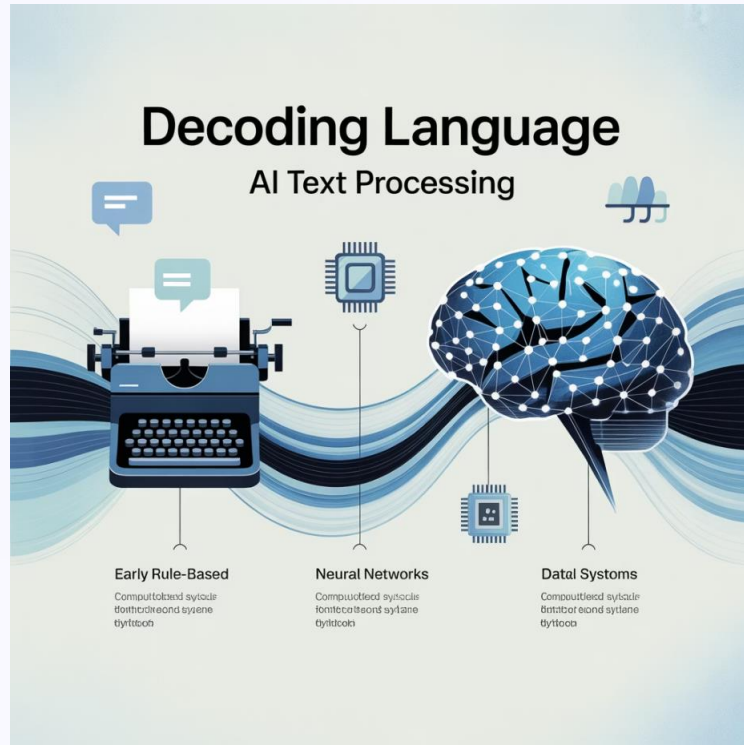
04 Named Entity Recognition

Виявлення імен людей, місць, організації, дат

05 Семантичний аналіз

Розуміння смислу та контексту текстових даних

Архітектури ШІ для обробки тексту



Традиційні методи

Правила, словники, статистичні підходи - швидкі, але обмежені

Глибинне навчання

RNN, LSTM - розуміння послідовностей та контексту

1

2

3

4

Машинне навчання

SVM, Random Forest, Naive Bayes - ефективні для класифікації

Трансформери

BERT, GPT - революційний прорив у розумінні мови

Практичний приклад: NLTK для обробки тексту

Токенізація

Розділення речень на окремі слова та символи

```
import nltktokens =  
nltk.word_tokenize(text)
```

POS-тегінг

Визначення частин мови для кожного слова

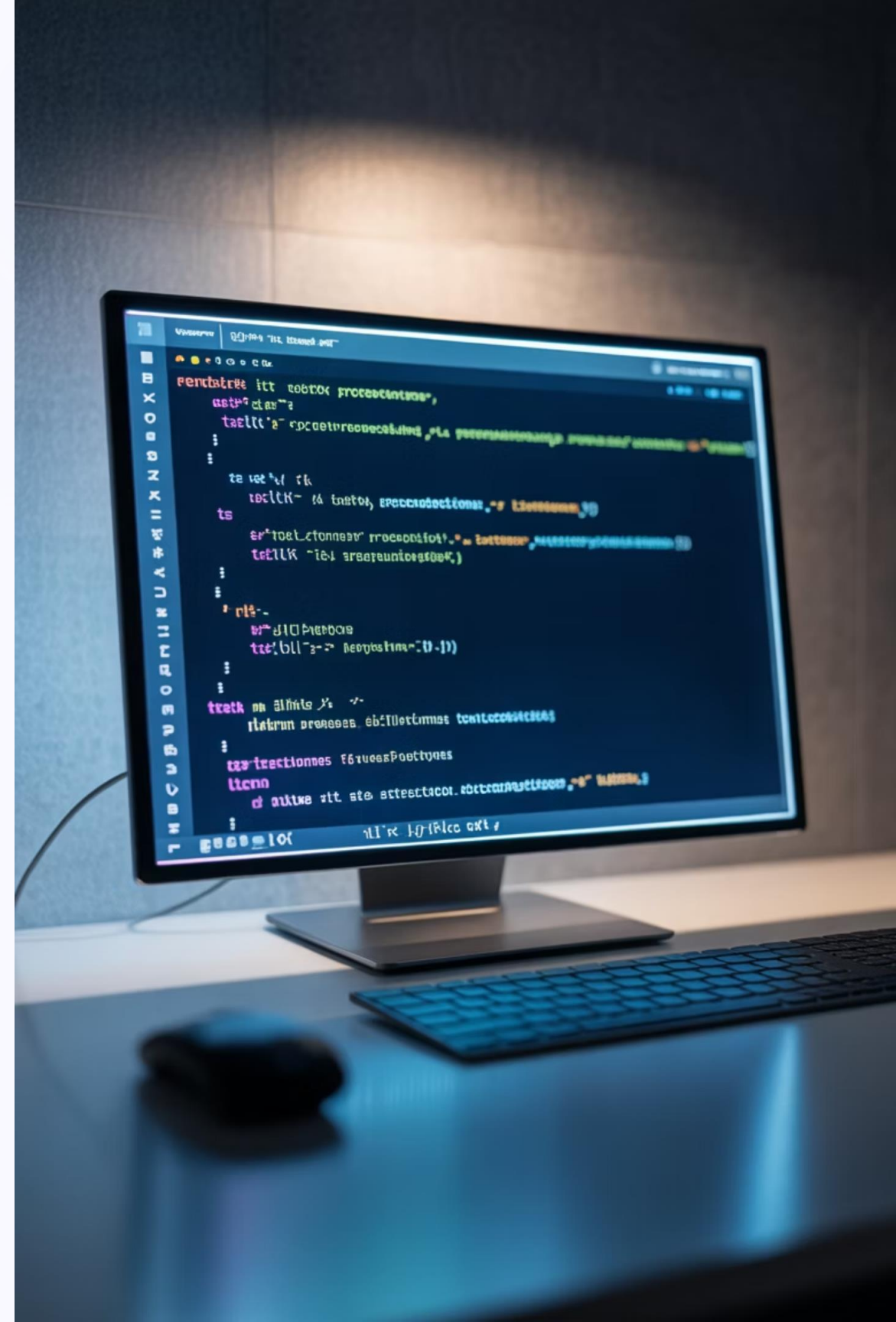
```
pos_tags = nltk.pos_tag(tokens)
```

Named Entity Recognition

Виявлення іменованих сутностей в тексті

```
entities = nltk.ne_chunk(pos_tags)
```

NLTK надає потужний інструментарій для базової обробки тексту українською та іншими мовами.



Використання трансформерів для аналізу текстів



BERT

Bidirectional Encoder - розуміє контекст слів у двох напрямках для точного аналізу



GPT

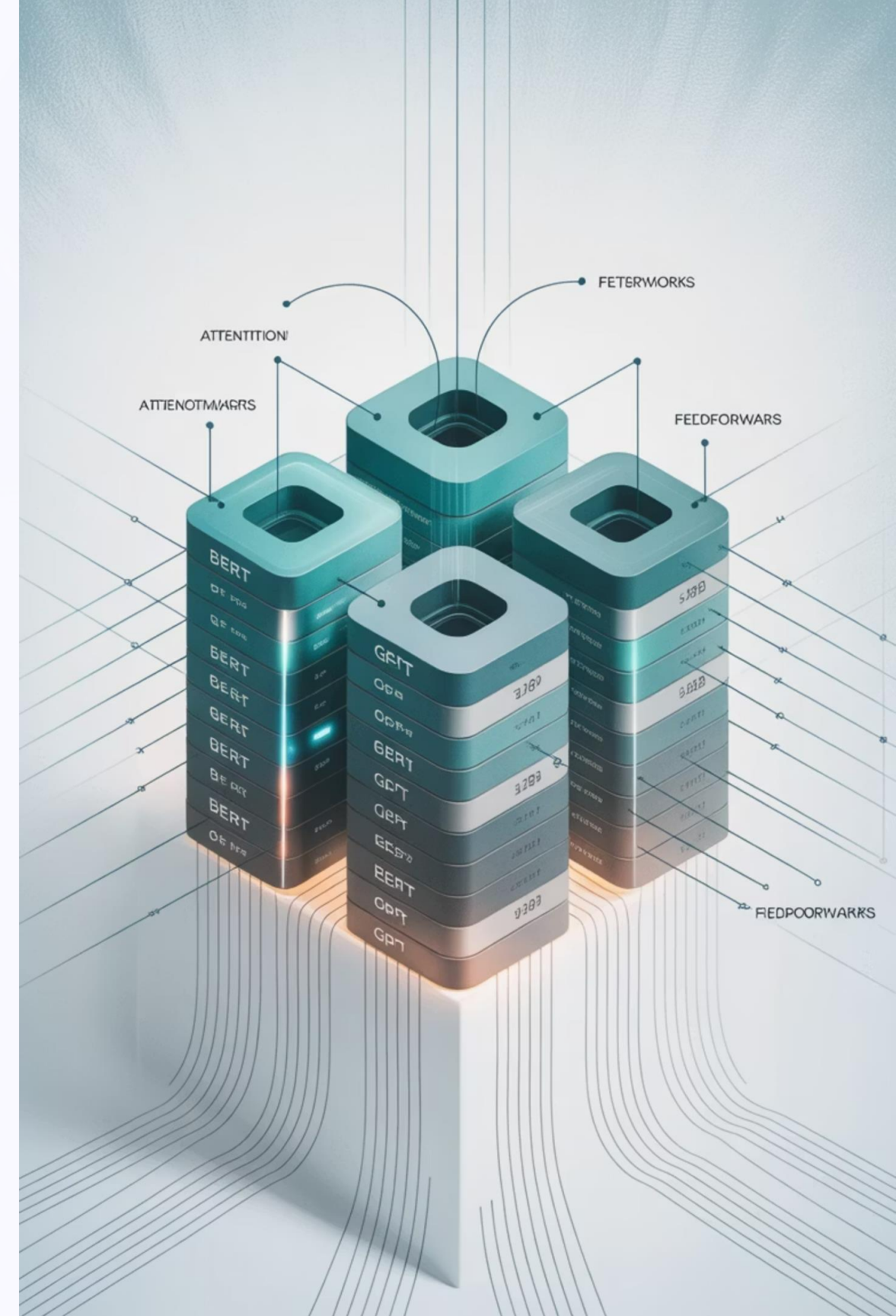
Generative Pre-trained - створює природний текст та відповідає на складні запити



Застосування

Автоматичне резюмування, Q&A системи, переклад та аналіз документів

Трансформери досягають точності понад 95% у багатьох задачах NLP, наближаючись до рівня людини.



Практична задача: Класифікація тональності

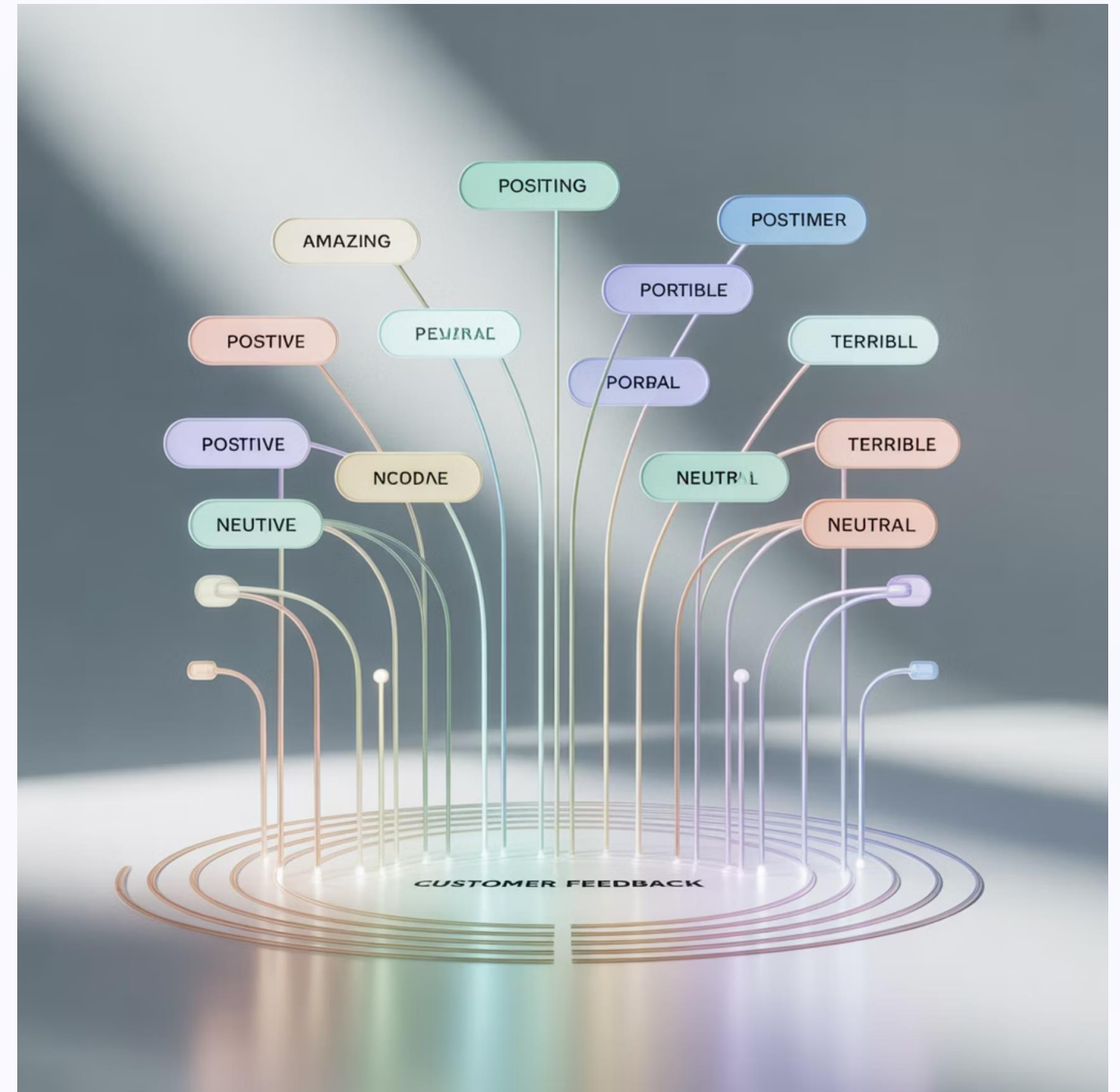
Постановка задачі

- Автоматичне визначення емоційного забарвлення
- Три класи: позитивний, негативний, нейтральний
- Обробка відгуків клієнтів та соціальних мереж

Оцінка якості

- Точність (Accuracy)
- F1-міра для збалансованої оцінки
- Confusion Matrix для детального аналізу

```
from transformers import pipeline
classifier = pipeline("sentiment-analysis")
result = classifier("Цей товар чудовий!")
# {'label': 'POSITIVE', 'score': 0.9998}
```



Named Entity Recognition

Named Entity Recognition (NER)



Імена людей

Автоматичне виявлення особистостей у текстах для аналітики та індексації



Організації

Розпізнавання компанії, установ та брендів для бізнес-аналітики



Географічні місця

Визначення країн, міст та адрес для геоаналітики

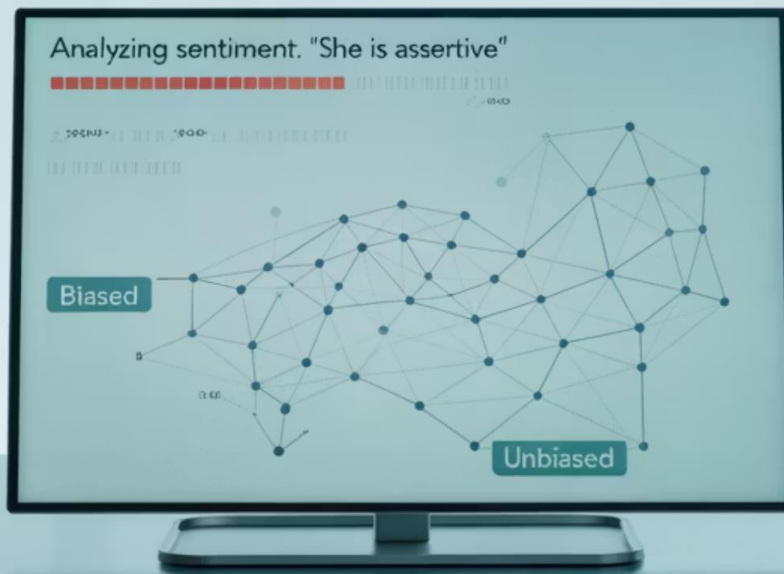


Дати та час

Вилучення темпоральної інформації для хронологічного аналізу

Інструменти spaCy та Stanford NER забезпечують точність понад 90% для української мови.

Виклики та проблеми аналізу текстів



Полісемія та контекст

Одне слово може мати різні значення залежно від контексту використання

Багатомовність

Особливості граматики, синтаксису та культурних нюансів різних мов

Етичні питання

Упередження моделей, приватність даних, справедливість алгоритмів

Контроль якості

Необхідність людської перевірки критично важливих рішень ШІ

Інструменти та платформи для програмування

Python бібліотеки

NLTK, spaCy, Gensim
- для базової
обробки. Hugging
Face Transformers -
для сучасних
моделей

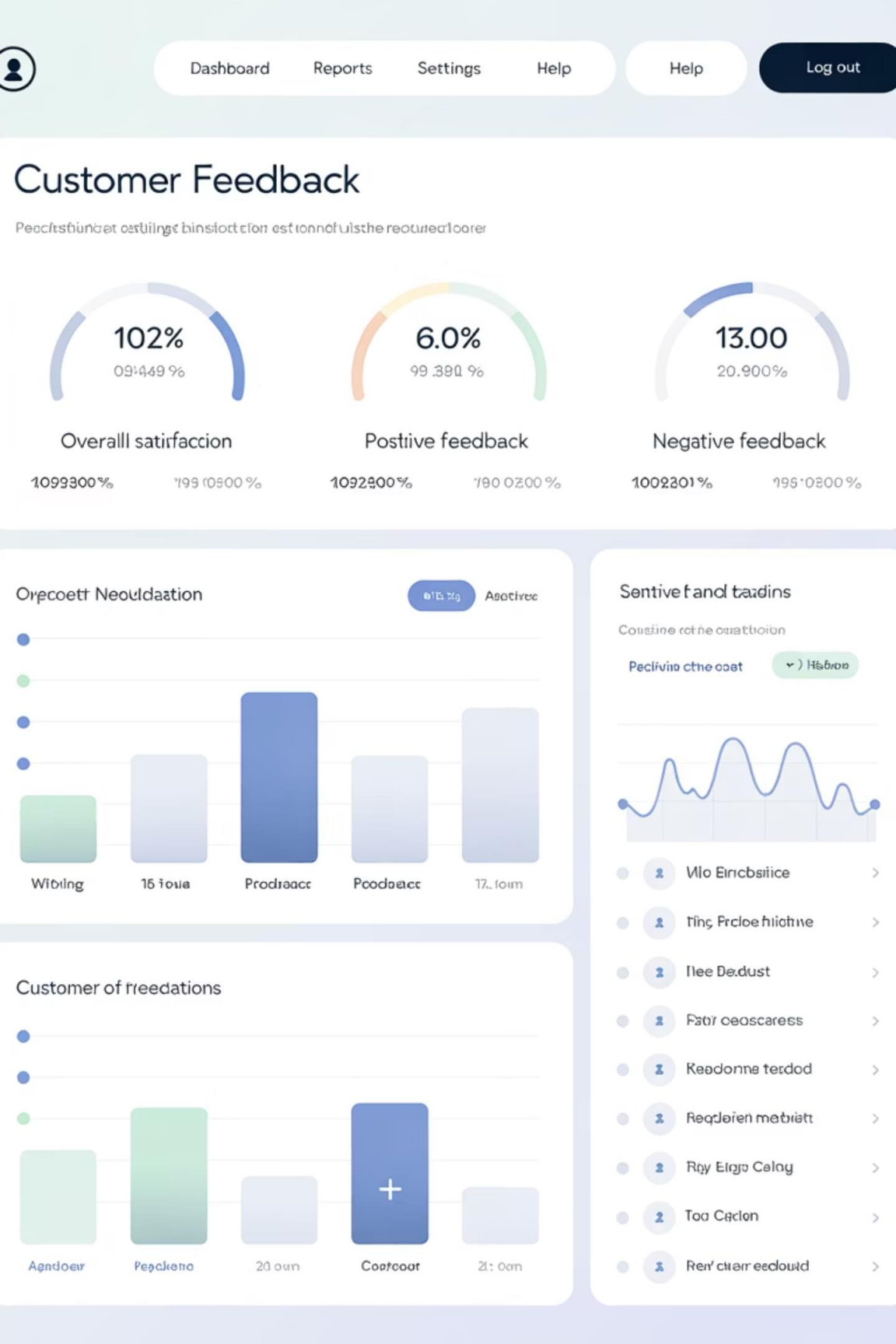
Хмарні сервіси

Google Cloud NLP,
AWS Comprehend,
Azure Text Analytics -
готові API з високою
точністю

ML фреймворки

TensorFlow, PyTorch
- для створення
власних архітектур
глибинного
навчання

❏ **Рекомендація:** Початківцям варто почати з spaCy та Hugging Face для швидкого старту



Комплексне рішення: аналіз відгуків клієнтів

Збір даних

Автоматичний збір відгуків із соціальних мереж, сайтів та додатків за допомогою API

Попередня обробка

Очищення від шуму, нормалізація тексту, видалення спаму та дублікатів

ШІ-аналіз

Класифікація тональності, виявлення ключових тем та проблемних зон

Візуалізація та звіти

Створення інтерактивних дашбордів для менеджменту з рекомендаціями

Перспективи розвитку: майбутнє аналізу текстів

Мультиmodalність

Поєднання тексту з зображеннями та відео для глибшого розуміння контексту

Реальний час

Миттєвий аналіз потоків даних для оперативного прийняття рішень



Пояснювальний ШІ

Прозорі алгоритми, які можуть пояснити свої рішення людською мовою

Автогенерація

Створення якісного контенту: статті, звіти, креативні тексти

Відкритість

Спільні дослідження, відкриті датасети та демократизація ШІ

Підсумки

Ключова галузь

Аналіз текстів із ШІ стає основою сучасних цифрових систем та бізнес-процесів

Гібридний підхід

Найкращі результати досягаються поєднанням класичних методів і глибокого навчання

Доступність інструментів

Сучасні бібліотеки та API дозволяють швидко створювати практичні рішення

Відповідальність

Етичні та якісні аспекти залишаються критично важливими для успіху проектів



**Unlock insights
from your data with
intelligent text analysis**



Sentiment
analysis



Topic
extraction



Keyword
detection

Контрольні питання

- 1 Які основні етапи обробки тексту в задачах аналізу?
- 2 Чим відрізняються традиційні методи від глибинного навчання?
- 3 Що таке токенізація і навіщо вона потрібна?
- 4 Як працює модель BERT?
- 5 Які інструменти можна використовувати для NER?
- 6 Які проблеми виникають при аналізі багатомовних текстів?
- 7 Як оцінити якість класифікації текстів?
- 8 Які етичні питання пов'язані з використанням ШІ в аналізі текстів?
- 9 Наведіть приклад практичного застосування аналізу текстів у бізнесі.
- 10 Які перспективи розвитку технології аналізу текстів?





Дякую за увагу!

Питання та обговорення