Практичне завдання №4: **Структуроване вилучення даних** (Data Extraction)

Мета: Навчитись створювати промпти, які здатні знаходити, вилучати та структурувати конкретну інформацію з тексту, форматуючи її у вигляді JSON.

Ключові концепції: Zero-Shot Prompting, Few-Shot Prompting, форматування виводу, визначення схеми даних (schema).

Завдання 1: **Парсинг опису товару** (Zero-Shot)

Ідея: Витягти ключові характеристики товару з відносно простого та структурованого тексту.

Завдання:

1. Оберіть (скопіюйте) 2-3 опису товару однакової категорії з будь якої площадки онлайн-торгівлі. Вибір категорії товару та відповідних полів (якщо це не категорія ноутбуків) залишається на ваш власний розсуд (підгодьте творчо до даних аспектів).
2. Напишіть промпт, який вилучає з обраного тексту інформацію про товар і представляє її у форматі JSON з чітко визначеними полями.

Вхідний текст (**приклад**):

Ноутбук "CosmoBook Pro 15", найновіша модель 2024 року. Оснащений 15.6-дюймовим OLED-дисплеєм, процесором Intel Core i9, 32 ГБ оперативної пам'яті та SSD на 1 ТБ. Вага пристрою складає всього 1.8 кг. Ціна — 75 999 грн. На складі залишилось 15 штук.

Промпт (**приклад**):

```
Вилучи з наведеного нижче тексту ключову інформацію про товар.
Представ результат у форматі JSON.

Схема JSON повинна містити такі поля:
- "product_name": назва товару (string)
- "model_year": рік випуску (integer)
- "screen_size_inches": діагональ екрану в дюймах (float)
- "ram_gb": обсяг оперативної пам'яті в ГБ (integer)
- "storage_tb": обсяг SSD в ТБ (float)
- "weight_kg": вага в кг (float)
- "price_uah": ціна в гривнях (integer)
- "in_stock": кількість на складі (integer)
```

```
Текст для аналізу:
"""
Ноутбук "CosmoBook Pro 15", найновіша модель 2024 року.
Оснащений 15.6-дюймовим OLED-дисплеєм, процесором Intel Core i9,
32 ГБ оперативної пам'яті та SSD на 1 ТБ. Вага пристрою складає
всього 1.8 кг. Ціна — 75 999 грн. На складі залишилось 15 штук.
"""
```

Оцініть:

1. Наскільки точно модель впоралася із завданням? Чи всі поля заповнені правильно?
2. Чи правильно модель визначила типи даних (наприклад, 75999 як integer, а не string)?
3. Що станеться, якщо якесь поле у тексті відсутнє (наприклад, не вказано вагу)? Як модель себе поведе? (Можна спробувати видалити частину тексту і перевірити).

Завдання 2: **Парсинг новинної статті** (**більш складний текст**)

Ідея: Ускладнимо завдання, використовуючи менш структурований текст — коротку новину.

Завдання:

1. Оберіть (скопіюйте) 2-3 коротких новини (або фрагментів новин) з будь-якого медіа-порталу чи онлайн-видання (ЗМІ). Вибір тематики новин на ваш власний розсуд (підгодьте творчо до даного аспекту).
2. Створіть промпт для вилучення ключових сутностей з новинної замітки.

Вхідний текст (**приклад**):

Київ, 21 вересня — Технологічний гігант "FutureTech" оголосив про відкриття нового дослідницького центру в інноваційному парку UNIT.City. Захід відвідав CEO компанії, Іван Петренко. Компанія планує інвестувати 5 мільйонів доларів у розробку штучного інтелекту та створити понад 200 нових робочих місць до кінця 2025 року.

Промпт (**приклад**):

```
Проаналізуй текст новини та вилучи з нього ключові сутності.
Відформатуй відповідь як JSON об'єкт.
```

```
Якщо якась інформація відсутня, використовуй значення null.

JSON схема:
- "organization": назва компанії (string)
- "person": згадана особа (string)
- "location": місто або конкретне місце (string)
- "investment_amount_usd": сума інвестицій в доларах (integer)
- "new_jobs_count": кількість нових робочих місць (integer)
- "deadline_year": рік, до якого планують реалізувати плани
(integer)

Текст новини:
"""
Київ, 21 вересня – Технологічний гігант "FutureTech" оголосив
про відкриття нового дослідницького центру в інноваційному парку
UNIT.City. Захід відвідав CEO компанії, Іван Петренко. Компанія
планує інвестувати 5 мільйонів доларів у розробку штучного
інтелекту та створити понад 200 нових робочих місць до кінця
2025 року.
"""
```

Оцініть:

1. Чи змогла модель правильно ідентифікувати всі сутності?
2. Як модель впоралася з числовими даними ("5 мільйонів" -> 5000000)?
3. Чи є у відповіді зайва інформація, яка не відповідає схемі? Як цього уникнути?

Завдання 3: **Підвищення надійності за допомогою прикладу** (Few-Shot Prompting)

Ідея: Продемонструвати, як надання прикладу може покращити якість та стабільність результату, особливо при роботі зі складними або неоднозначними текстами.

Завдання:

1. Модифікуйте попередній промпт, додавши до нього один приклад обробки тексту ("few-shot").

Приклад промпту (**доповнений**):

Ти — експерт з вилучення структурованої інформації з тексту. Твоє завдання — перетворити текст на JSON. Дотримуйся наданої схеми та прикладу.

Приклад:
Текст: "Компанія AeroDynamics з Дніпра отримала грант на 200 тисяч євро від European Space Agency для розробки нового дрона."
JSON:
```
{
  "organization": "AeroDynamics",
  "location": "Дніпро",
  "grant_amount_eur": 200000,
  "source_organization": "European Space Agency",
  "product": "дрон"
}
```

Завдання:
Проаналізуй текст новини та вилучи з нього ключові сутності. Відформатуй відповідь як JSON об'єкт.

JSON схема:
- "organization": назва компанії (string)
- "person": згадана особа (string)
- "location": місто або конкретне місце (string)
- "investment_amount_usd": сума інвестицій в доларах (integer)
- "new_jobs_count": кількість нових робочих місць (integer)
- "deadline_year": рік, до якого планують реалізувати плани (integer)

Текст новини:
"""
Київ, 21 вересня – Технологічний гігант "FutureTech" оголосив про відкриття нового дослідницького центру в інноваційному парку UNIT.City. Захід відвідав CEO компанії, Іван Петренко. Компанія планує інвестувати 5 мільйонів доларів у розробку штучного

```
інтелекту та створити понад 200 нових робочих місць до кінця
2025 року.
"""
```

Оцініть:

1. Чи змінився результат після додавання прикладу? Чи став він більш точним або стабільним при повторних запусках?
2. Як приклад допомагає моделі краще зрозуміти контекст та очікуваний формат?
3. В яких випадках техніка Few-Shot є найбільш корисною?
4. Чи змогла модель правильно ідентифікувати всі сутності?

Завдання 4: **Подумайте над питаннями**

- **З якими основними труднощами ви зіткнулися при створенні промптів для парсингу**? (напр., неоднозначність тексту, пропущена інформація, складні формати чисел).
- **Як можна було б покращити промпти** для обробки більш складних випадків (наприклад, коли в тексті згадується кілька компаній або осіб)?
- **Подумайте про реальні кейси застосування такого "міні-парсера"** у вашій майбутній професійній діяльності.