2. Практическое занятие: ресурсы текстов и корпусы, этапы создания языкового корпуса

Инструментарий: Python 3.x (модуль `re`)

Введение и теоретический обзор

Создание корпуса — это не просто сбор текстов. Это процесс превращения «сырых» данных в структурированный ресурс, пригодный для машинного обучения или лингвистических исследований.

Основные этапы создания корпуса:

1. **Сбор данных (Scraping/Collection):** Получение текстов из источников.
2. **Очистка (Cleaning):** Удаление технического «шума» (HTML, спецсимволы).
3. **Аннотирование (Annotation):** Добавление метаданных (автор, дата) или меток (тональность, тема).

Задание 1. Постановка цели и проектирование

Прежде чем писать код, необходимо определить границы исследования.

Инструкция:

Заполните «Паспорт корпуса», ответив на вопросы:

- **Зачем?** (Например: классификация спама, анализ отзывов).
- **На каком языке?** (Например: узбекский, русский, многоязычный).
- **Тип текстов?** (Например: официальные документы, посты из Telegram).

Результат: Заполненная карточка проекта (3–5 предложений).

Задание 2. Выбор источников и инструментов

Изучите таблицу популярных ресурсов и выберите подходящие для вашей цели.

Тип источника	Пример ресурса	Инструмент Python
Web-страницы	Wikipedia, Common Crawl	`BeautifulSoup`, `Scrapy`

Тип источника	Пример ресурса	Инструмент Python
Готовые наборы	Hugging Face Datasets	`datasets` library
Соцсети	Twitter (X), Reddit	API или библиотеки-парзеры

Задание: Составьте список из 2–3 конкретных сайтов или платформ, откуда вы могли бы собрать данные.

Задание 3. Программная подготовка (Сбор данных)

На этом этапе мы имитируем получение данных, создавая список строк.

Инструкция:

Запустите среду (Jupyter Notebook или VS Code) и инициализируйте первичные данные.

Python
```
# Исходные "сырые" данные с шумом (HTML-теги, лишние знаки, цифры)
raw_data = [
    "Bu birinchi matn. <br> NLP haqida 100%!",
    "Ikkinchi matn!!!   Ortiqcha belgilar bor...",
    "Uchinchi matn 😊 va raqamlar 2024"
]
print("Данные получены:", len(raw_data), "записей.")
```

Задание 4. Очистка данных (Preprocessing)

Компьютеру сложно обрабатывать слова «Текст», «текст!» и «текст123» как одно и то же слово. Нам нужно привести их к единому стандарту.

Инструкция:

Используйте регулярные выражения (re) для фильтрации.

Python
```
import re

def clean_text(text):
    # 1. Нижний регистр
    text = text.lower()
    # 2. Удаление HTML-тегов
```

```
    text = re.sub(r"<.*?>", " ", text)
    # 3. Удаление цифр и спецсимволов (оставляем буквы и пробелы)
    # Регулярное выражение адаптировано под кириллицу и узбекскую латиницу
    text = re.sub(r"[^a-za-яёўқғҳ\s]", "", text)
    # 4. Удаление лишних пробелов
    text = re.sub(r"\s+", " ", text).strip()
    return text

clean_texts = [clean_text(t) for t in raw_data]
print("Очищенный результат:", clean_texts)
```

Задание 5. Структурирование корпуса (JSON-style)

Корпус должен содержать не только текст, но и **метаданные**. Мы будем использовать формат «список словарей» (похож на JSON).

Инструкция:

Превратите плоский список строк в структурированный объект.

Python
```
corpus = []

for i, text in enumerate(clean_texts):
    entry = {
        "id": i + 1,
        "text": text,
        "metadata": {
            "language": "uz",
            "source": "manual_entry",
            "is_cleaned": True
        }
    }
    corpus.append(entry)

# Просмотр первой записи
print(corpus[0])
```

Задание 6. Базовая разметка (Annotation)

Разметка — это присвоение тексту категории.

Инструкция:

Добавьте ключ label к вашим записям, определив их тематику.

Python
```
# Пример ручной разметки
```

```
labels = ["education", "grammar", "general"]

for i in range(len(corpus)):
    corpus[i]["label"] = labels[i]

print("Размеченная запись:", corpus[0])
```

Контрольные вопросы для самопроверки

1. Почему важно переводить текст в нижний регистр перед анализом частоты слов?
2. В каких случаях **нельзя** удалять цифры из корпуса? (Пример: финансовая аналитика).
3. Что такое «шум» в контексте текстовых данных?
4. Чем формат словаря удобнее для корпуса, чем обычный список строк?

Итог работы: Вы создали мини-корпус, который готов для сохранения в файл (например, `.json` или `.csv`) и дальнейшего использования в моделях машинного обучения.