### **Лабораторна робота 11**

### **1.1. Токенізація**

Токенізація — це процес розбиття тексту на менші одиниці, звані токенами (слова, фрази, символи або інші елементи). Токенізація є першим етапом обробки тексту перед виконанням подальших операцій.

* **Приклад**: "Привіт, як справи?" → ["Привіт", ",", "як", "справи", "?"]

### **1.2. Лемматизація та стемінг**

* **Стемінг** — процес скорочення слова до його основи (стема), що може не бути правильним словом. Наприклад, "running" → "run".
* **Лемматизація** — процес зведення слова до його базової або початкової форми (леми), яка є правильним словом у мові. Наприклад, "better" → "good".

### **1.3. Векторизація тексту**

Векторизація тексту дозволяє перетворювати текст у числові формати, які можуть бути використані для машинного навчання.

* **Bag of Words (BoW)** — представлення тексту як набору частот слів, без врахування порядку слів.
* **TF-IDF (Term Frequency-Inverse Document Frequency)** — метод, який оцінює важливість слова в документі та серед всіх документів.
* **Word Embeddings** — це методи, що використовують векторні представлення слів (наприклад, Word2Vec, GloVe), що дозволяють моделювати семантичні зв'язки між словами.

### **1.4. Класифікація тексту**

Класифікація тексту полягає в призначенні певної категорії тексту. Це може бути, наприклад, категоризація електронних листів на "спам" або "не спам".

* Алгоритми: наївний баєсовий класифікатор, логістична регресія, дерева рішень.

### **1.5. Розпізнавання сутностей (Named Entity Recognition, NER)**

NER — це процес автоматичного виявлення та категоризації сутностей в тексті, таких як імена людей, організації, місця, дати тощо.

* Приклад: "Барак Обама був обраний президентом США в 2008 році." → "Барак Обама" (персона), "США" (країна), "2008" (рік).

### **1.6. Ключові моделі для NLP**

* **Наївний баєсовий класифікатор** — ймовірнісна модель, що базується на байєсівському теоремі для класифікації текстів.
* **Логістична регресія** — метод класифікації, який використовує логістичну функцію для визначення ймовірності приналежності тексту до певного класу.
* **LSTM (Long Short-Term Memory)** — тип рекурентної нейронної мережі, яка здатна обробляти послідовності даних, зберігаючи інформацію на тривалий час.
* **Transformers** — архітектура нейронних мереж, яка дозволяє ефективно обробляти довгі послідовності та є основою для таких моделей, як BERT, GPT.
* **GPT (Generative Pretrained Transformer)** — модель, основана на трансформерах, яка генерує текст на основі заданого контексту.

### **2. Порівняльний аналіз методів векторизації**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Метод** | **Переваги** | **Недоліки** | **Складність реалізації** | **Застосування** | **Складність обробки великих обсягів даних** |
| **Bag of Words (BOW)** | Простий та швидкий метод. | Ігнорує порядок слів. | Легка реалізація. | Класифікація тексту. | Легко обробляти, але стає важким з великими даними (великі матриці). |
|  | Легко інтегрується з різними моделями. | Відсутність семантичної інформації. |  | Виявлення тем в тексті. | Потрібен великий обсяг пам'яті для зберігання великої кількості ознак. |
| **TF-IDF** | Враховує важливість слів у контексті всього корпусу. | Все ще ігнорує порядок слів. | Легка реалізація. | Класифікація тексту. | Легко обробляти, але стає складнішим з великими даними через високий розмір матриць. |
|  | Може знижувати вагу загальних слів, таких як "the", "is". | Не враховує семантичні зв'язки між словами. |  | Виявлення важливих термінів. | Може бути повільним на великих наборах даних. |
| **Word Embeddings** | Враховує семантичні зв'язки між словами. | Складніша реалізація, потребує значних обчислювальних ресурсів. | Складна реалізація. | Глибоке навчання моделей. | Велика кількість даних вимагає значних ресурсів для тренування. |
| (Word2Vec, GloVe) | Може створювати більш компактні вектори. | Тренування моделей займає час і ресурси. |  | Пошук схожих слів. | Потрібен великий обсяг пам'яті для зберігання моделей. |
|  | Добре працює з великими обсягами даних. | Може бути недостатньо точним для рідкісних слів. |  | Генерація контекстуальних векторів. | Високі вимоги до обчислювальних потужностей. |

**3.1. Огляд інструментів дял NLP:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Інструмент** | **Основні функції** | **Підтримка мов** | **Простота використання** | **Особливості** | **Приклади застосувань в NLP** |
| **NLTK (Natural Language Toolkit)** | - Токенізація, стемінг, лемматизація, парсинг, класифікація, пошук сутностей | Багато мов, в основному англійська | Легка для початківців, має багато інструментів для навчання | - Добре для навчання та експериментів. | - Аналіз тональності (sentiment analysis).  - Класифікація тексту.  - Пошук сутностей. |
| **SpaCy** | - Токенізація, лемматизація, розпізнавання сутностей (NER), залежність між словами, векторизація | Підтримує багато мов (англійська, німецька, французька, іспанська тощо) | Простий API, оптимізовано для швидкої роботи | - Орієнтований на виробничі системи.  - Висока продуктивність.  - Підтримує статистичне навчання. | - Чат-боти.  - Класифікація тексту.  - Пошук сутностей (NER). |
| **Hugging Face Transformers** | - Моделі трансформерів (BERT, GPT, T5 тощо), тренування та застосування трансформерів | Підтримує багато мов, спеціалізується на англійській | Легко використовувати для високопродуктивних задач, великий набір моделей | - Підтримує найновіші моделі NLP.  - Призначено для глибокого навчання.  - Велика спільнота та документація. | - Чат-боти.  - Генерація тексту.  - Машинний переклад.  - Аналіз тональності. |
| **Gensim** | - Векторизація тексту (Word2Vec, Doc2Vec), тематичне моделювання, векторні простори | Підтримує англійську та інші мови | Легка в використанні для завдань тематичного моделювання | - Орієнтований на моделювання тем та векторні представлення текстів.  - Хороший для великих текстових корпусів. | - Тематичне моделювання.  - Пошук схожих документів.  - Рекомендаційні системи. |

### **3.2 Приклади застосувань NLP в різних галузях:**

1. **Аналіз тональності**: Використовується для аналізу емоційної забарвленості текстів, таких як відгуки користувачів або твітти.
2. **Чат-боти**: Моделі, такі як GPT або трансформери, використовуються для створення чат-ботів, які можуть ефективно взаємодіяти з користувачами.
3. **Рекомендаційні системи**: Використання тематичного моделювання (наприклад, за допомогою Gensim) для створення систем, які рекомендують контент на основі аналізу текстів користувачів.