Зображення, що містить текст

Автоматично згенерований опис

Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут імені Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

**Лабораторна робота №3**

Аналіз текстів з використанням мови Python

**Тема:** Моделі текстових даних

**Варіант:** 1

Виконав Перевірив:

студент групи ІП-11: Тимофєєва Ю. С

Панченко С. В.

Київ 2023

Зміст

[1 Мета лабораторної роботи 6](#__RefHeading___Toc189_2645281070)

[2 Завдання 7](#__RefHeading___Toc191_2645281070)

[3 Виконання 8](#__RefHeading___Toc193_2645281070)

[3.1 Представити корпус як модель «Сумка слів». Вивести вектор для слова «film». 8](#__RefHeading___Toc195_2645281070)

[3.2 Представити корпус як модель TD-IDF. Спробувати кластеризувати документи за допомогою ієрархічної агломераційної кластеризації. 10](#__RefHeading___Toc197_2645281070)

[3.3 Представити корпус як модель Word2Vec. Знайти подібні слова до слів shrimp, economy. 12](#__RefHeading___Toc199_2645281070)

[4 Висновок 14](#__RefHeading___Toc201_2645281070)

# Мета лабораторної роботи

Ознайомитись з основними текстовими моделями та їх створення за допомогою scikit-learn та genism.

# Завдання

Варіант 1.

Зчитати файл doc1. Вважати кожен рядок окремим документом

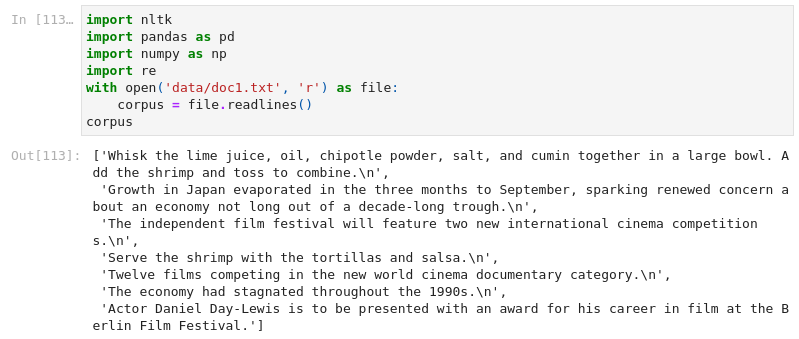
корпусу. Виконати попередню обробку корпусу.

1. Представити корпус як модель «Сумка слів». Вивести вектор для слова «film».
2. Представити корпус як модель TD-IDF. Спробувати кластеризувати документи за допомогою ієрархічної агломераційної кластеризації.
3. Представити корпус як модель Word2Vec. Знайти подібні слова до слів shrimp, economy.

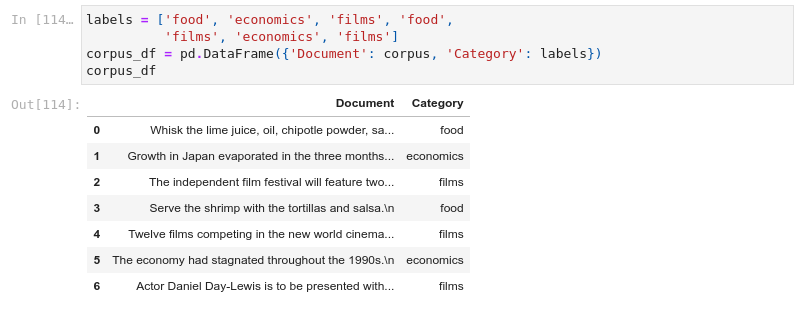
# Виконання

## Представити корпус як модель «Сумка слів». Вивести вектор для слова «film».

Зчитаємо текст з файлу та імпортуємо модулі nltk, pandas, numpy, re.

  
  
Рисунок 3.1 - Зчитування файлу

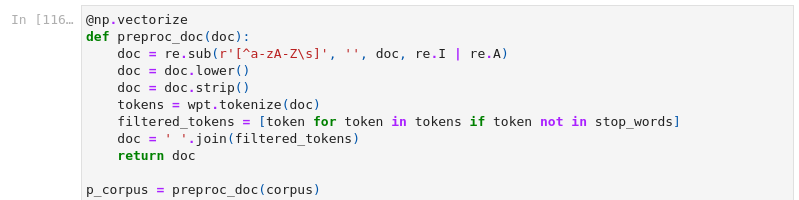
Задамо теми кожного документу.

  
  
Рисунок 3.2 - Теми документів

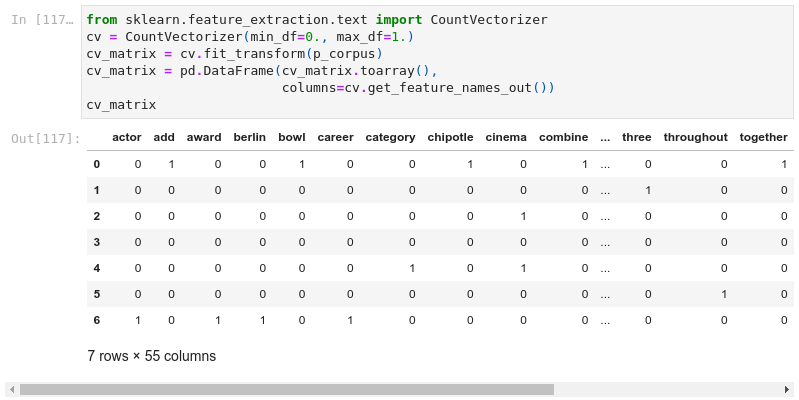
Визначимо стоп-слова англійської мови.

  
  
Рисунок 3.3 - Стоп-слова

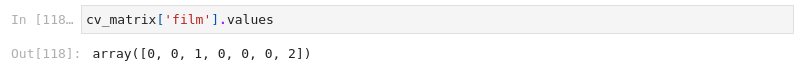
Визначимо функцію, що виконує попередню обробку документу. Застосуємо декоратор np.vectorize для того, щоб функція могла працювати з корпусами.

  
  
Рисунок 3.4 - Обробка документів

Представимо корпус як модуль "Сумка слів". Використаємо для цього клас CountVectorizer зі sklearn.feature\_extraction.text.

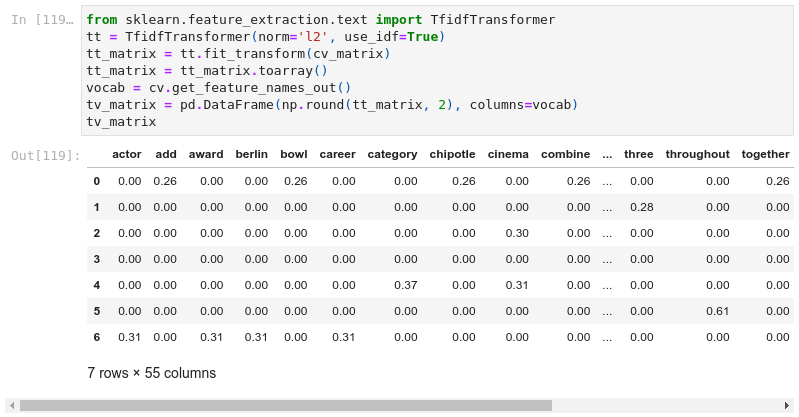
  
  
Рисунок 3.5 - Сумка слів

Виведемо вектор слова "film".

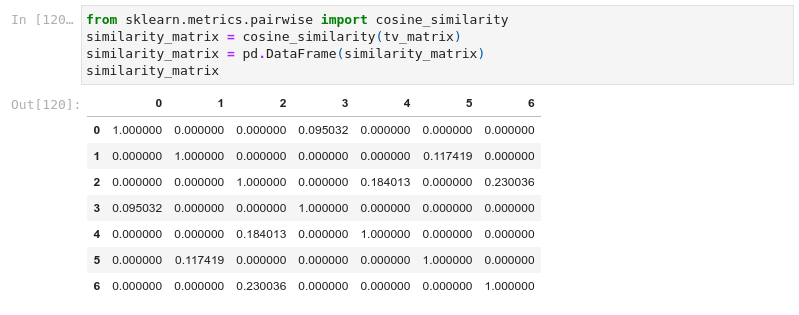
  
  
Рисунок 3.6 - Вектор слова "film"

## Представити корпус як модель TD-IDF. Спробувати кластеризувати документи за допомогою ієрархічної агломераційної кластеризації.

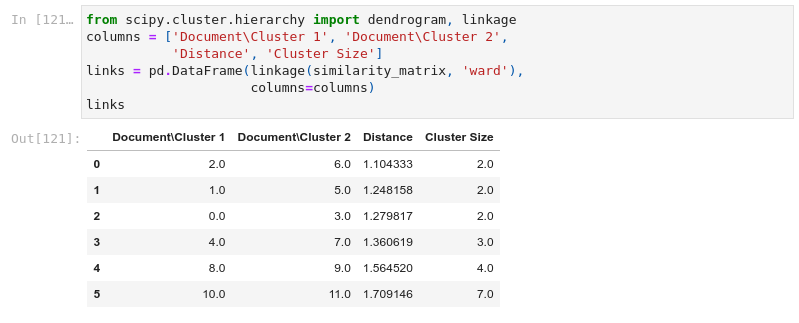
Перетворимо матрицю з частотою термінів на матрицю tfidf.

  
  
Рисунок 3.7 - Матриця TF-IDF

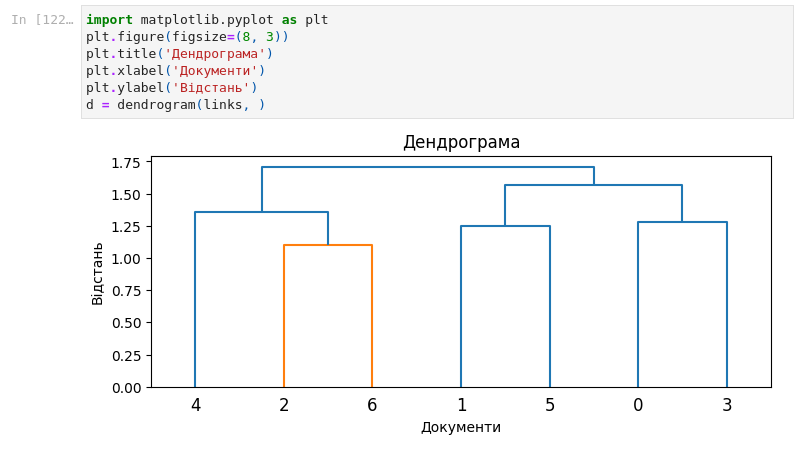
Обрахуємо подібність векторів за допомогою косинусної відстані.

  
  
Рисунок 3.8 - Матриця подібності

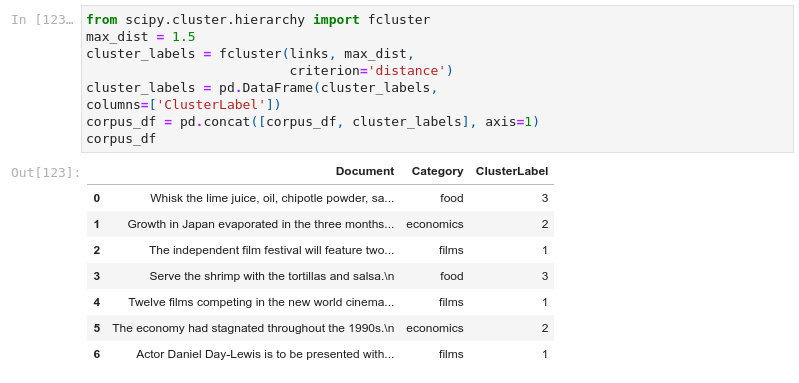
На основі матриці подібності побудуємо матрицю зв'язку.

  
  
Рисунок 3.9 - Матриця зв'язку

Візуалізуємо зв'язки за допомогою дендрограми.

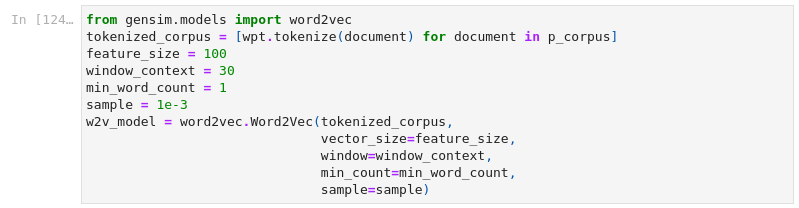
  
  
Рисунок 3.10 - Дендрограма

Отримаємо мітки кластерів. Бачимо, що кластери зливаються приблизно на відстані 1.5.

  
  
Рисунок 3.11 - Додавання міток

## Представити корпус як модель Word2Vec. Знайти подібні слова до слів shrimp, economy.

Побудуємо модель Word2Vec.

  
  
Рисунок 3.12 - Модель Word2Vec

Подивимося подібні слова до "shrimp", "economy".

  
  
Рисунок 3.13 - Виведення подібних слів

# Висновок

Під час виконання даної лабораторної роботи я ознайомився з основними текстовими моделями та їхнім створенням за допомогою scikit-learn та genism.

У першому завданні було створено модель “Сумка слів” та виведено вектор слова "film”.

У другому завданні побудували модель TD-IDF, визначено подібність слів за допомогою косинусної відстані, проведено кластеризацію.

У третьому завданні побудовано модель Word2Vec та виведено найбільш подібні слова до “shrimp” та “economy”.