

2024-11-12

Практика 5. Простые рекомендательные алгоритмы на основе веб-данных

1. Собрать с сайтов-агрегаторов базу рейтингов фильмов, сериалов, книг, игр, музыки, мест для путешествий, ресторанов и т.п. (выбрать любую 1 категорию)
2. База должна включать не менее 200-250 записей
3. Векторизовать базу, используя в качестве признаков тэги на исходном сайте и авторов/актеров
4. Разработать приложение, воспользовавшись готовой реализацией метода k -ближайших соседей ([1.6. Nearest Neighbors — scikit-learn 1.3.2 documentation](#)), которое должно принимать на вход список контента пользователя (просмотренных фильмов/сериалов, прочитанных книг и т.п.) предложить элементарный рекомендательных алгоритм: порекомендовать следующий фильм/сериал/книгу и т.п. на основе входной информации.
5. Предусмотреть возможность оценки качества рекомендации пользователем

Лабораторная работа № 4. Анализ и визуализация геоданных

1. Взять файл [metro.tsv](#), который содержит название станции Московского метро, ее широту и долготу
2. Воспользоваться библиотекой [folium](#), [Folium — Folium 0.18.0 documentation](#) для реализации интерактивной карты, которая показывает карту Москвы с нанесенными на ней точками — станциями метро.
3. По нажатию на точку должно выводиться название станции и ее координаты.
4. Цвет точки должен зависеть от того, в каком административном округе находится станция (взять информацию можно, например, отсюда: [Москва. Метро. Станции метро по административным округам Москвы. Достопримечательности](#))
5. Дополнительно по работе с folium можно прочесть руководство: [Основы Интерактивных карт / Хабр](#)

Лабораторная работа № 5. Анализ социальных сетей и библиотека NetworkX



NetworkX

Network Analysis in Python

Сайт: <https://networkx.org/>

Документация: [Reference — NetworkX 3.2.1 documentation](#)

Тutorial: [Tutorial — NetworkX 3.2.1 documentation](#)

Примеры: [Gallery — NetworkX 3.2.1 documentation](#)

Исходный код: [GitHub - networkx/networkx: Network Analysis in Python](#)

PyPI: [networkx · PyPI](#)

```
pip install networkx
```

Старая вводная статья: <https://habr.com/ru/post/125898/>

Постановка

1. Изучите галерею и примеры кода библиотеки NetworkX
2. Создайте и выведите при помощи функции `draw` граф со списком ребер `(1,2), (1,3), (2,3), (2,4), (3,4)`
3. Выведите все базисные циклы (`cycle_basis`) в графе куба (`hypercube_graph`) размерности 3. Найдите все циклы в ориентированном графе при помощи `simple_cycles` в ориентированном графе (`DiGraph`), где каждому исходному ребру соответствует два ориентированных.
4. При помощи двойного цикла создайте случайный граф на 10 вершинах, в котором любая пара вершин соединена с вероятностью $p = 0.1$
5. Создайте в цикле 1000 раз случайный граф на 100 вершинах (`erdos_renyi_graph`), выбирая вероятность `p_i` случайно в интервале

- (0.005, 0.03). Найдите размер `s_i` наибольшей компоненты связности (`connected_components`). Нанесите полученные точки (`p_i`, `s_i`) на график.
6. После этой разминки визуализируйте кластеры в VK, следуя статье [Введение в анализ социальных сетей на примере VK API / Хабр](#).