Домашнее задание 2

В данном задании вам предстоит попрактиковаться в обходе Веба и анализе графа интернет-страниц.

1 Основное задание (4 балла)

1. Скачайте подграф Википедии размером порядка 10^4 страниц. Всю информацию о страницах хранить не нужно, достаточно только исходящих ссылок на другие страницы Википедии, заголовка и небольшого текстового описания (про это ниже). При обходе нужно игнорировать ссылки на служебные страницы (File:, Talk:, Category:, Template:, Wikipedia: и др.), а также Main Page. Рекомендуется использовать фреймворк Scrapy, рассмотренный на занятиях.

   Начать можно с кода примера, приведенного на занятии, а также с этого туториала: `http://pybae.github.io/blog/2015/04/27/a-simple-introduction-to-scrapy/`.

   Следует рассматривать только ссылки из mw-body-content, желательно ограничиваться не более, чем 100 первыми ссылками. Для этого в LinkExtractor можно указать restrict_xpaths или restrict_css (xpath будет выглядеть примерно так: '(//div[@id="mw-content-text"]/*/a/@href)[position() < 100]').

   В качестве start_urls выберите произвольный набор из нескольких страниц.

2. Исходя из сохраненной информации о ссылках между статьями, постройте граф. Создавать граф и работать с ним в python рекомендуется с помощью библиотеки networkx.

3. Посчитайте PageRank для вершин полученного графа с помощью одной из доступных вам библиотечных функций. Выведите топ-10 страниц по убыванию значений PageRank в формате
   Title <pagerank>
   URL
   Snippet
   Таким образом, выводимая информация будет похожа на настоящую поисковую выдачу.

Что взять в качестве Snippet? Можете решать сами, я предлагаю такой вариант:

BeautifulSoup(response.xpath('//div[@id="mw-content-text"]/p[1]').extract_first(), "lxml").text[:255]+"..."

Для `https://en.wikipedia.org/wiki/Information_retrieval` получится *'Information retrieval (IR) is the activity of obtaining information resources relevant to an information need from a collection of information resources. Searches can be based on full-text or other content-based indexing. Information retrieval is the scie...'*

4. Поварьируйте параметр alpha, по умолчанию равный 0.85. Например, рассмотрите значения 0.95, 0.5, 0.3. Изменяется ли выводимый топ?

2 Дополнительные задания

5. Для полученного графа реализуйте вычисление PageRank с помощью алгоритма 3 из статьи `https://projecteuclid.org/download/pdf_1/euclid.im/1128530802`. Что можно сказать о сходимости алгоритма? Насколько отличаются полученные значения для топа от значений, полученных с помощью библиотечной функции? **(2 балла)**

6. Примените к полученному графу алгоритм HITS (например, отсюда `https://networkx.github.io/documentation/networkx-1.9.1/reference/generated/networkx.algorithms.link_analysis.hits_alg.hits.html`). Сравните топ-10 результатов по значениям авторитетности, хабовости и их среднего с топом по PageRank. Сделайте выводы о наблюдаемых различиях. **(1 балл)**

3 Требования

Решение должно состоять из программного кода и pdf с отчетом. Если аналитику графа делаете в ipython notebook, можно отчет оформить в нем же.

Без решенных пунктов 1-4 задание не оценивается.

В случае сдачи задания после дедлайна баллы умножаются на коэффициент $\frac{20-n}{20}$, где n — количество полных суток опоздания.