Введение в анализ данных: Анализ ссылок

Юля Киселёва
juliakiseleva@yandex-team.ru
Школа анализа данных

План на сегодня

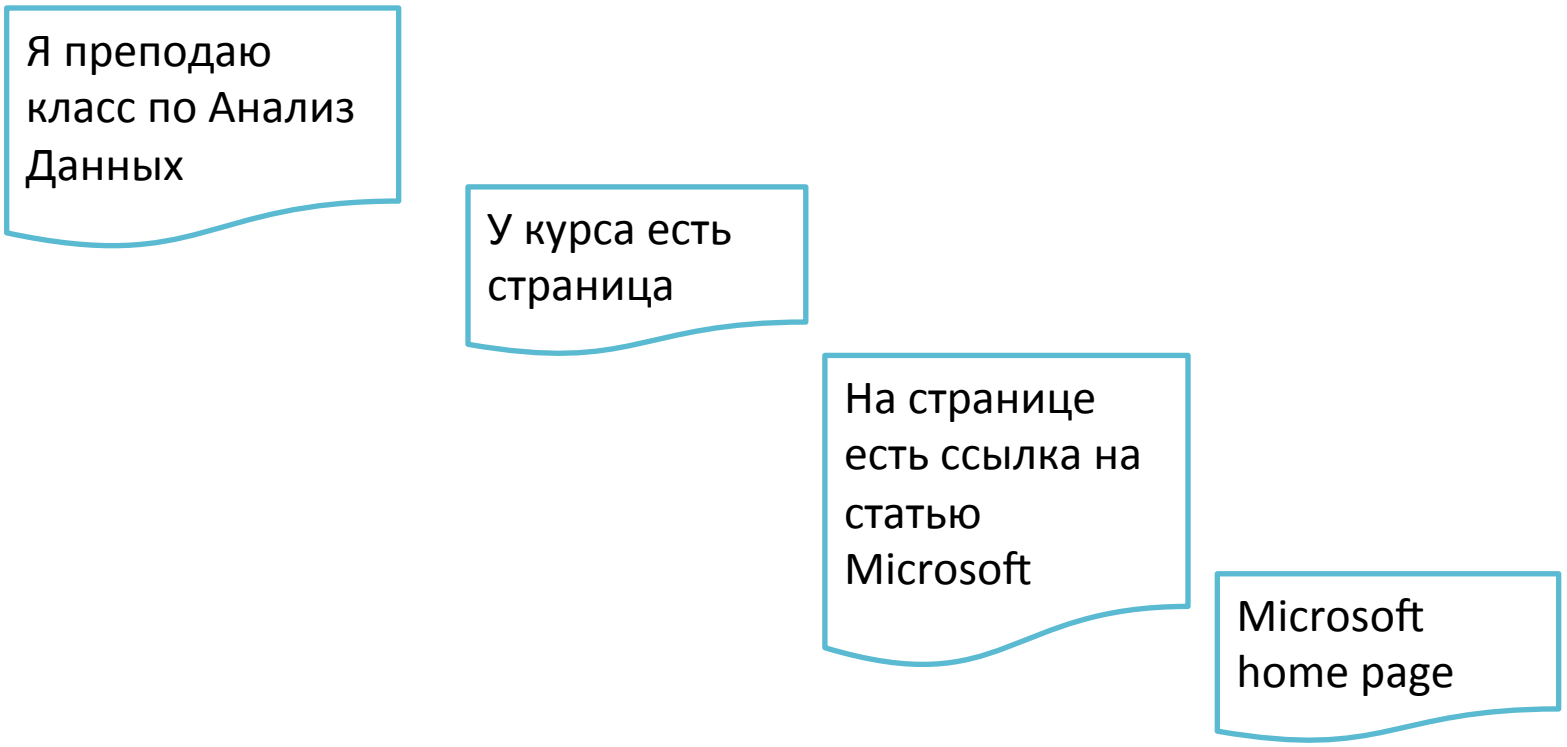
- Структура Интернета
- Page Rank

Насколько большой Интернет?

- Насколько большой интернет?
  - Технически, интернет бесконечный
  - Большая часть – это дубликаты (30-40%)
  - Соответственно наилучшая оценка сделана существующими поисковыми компаниями:
    - Google = 8 млрд страниц, Yahoo = 20 млрд страниц
- Какова структура интернета? Как он организован?

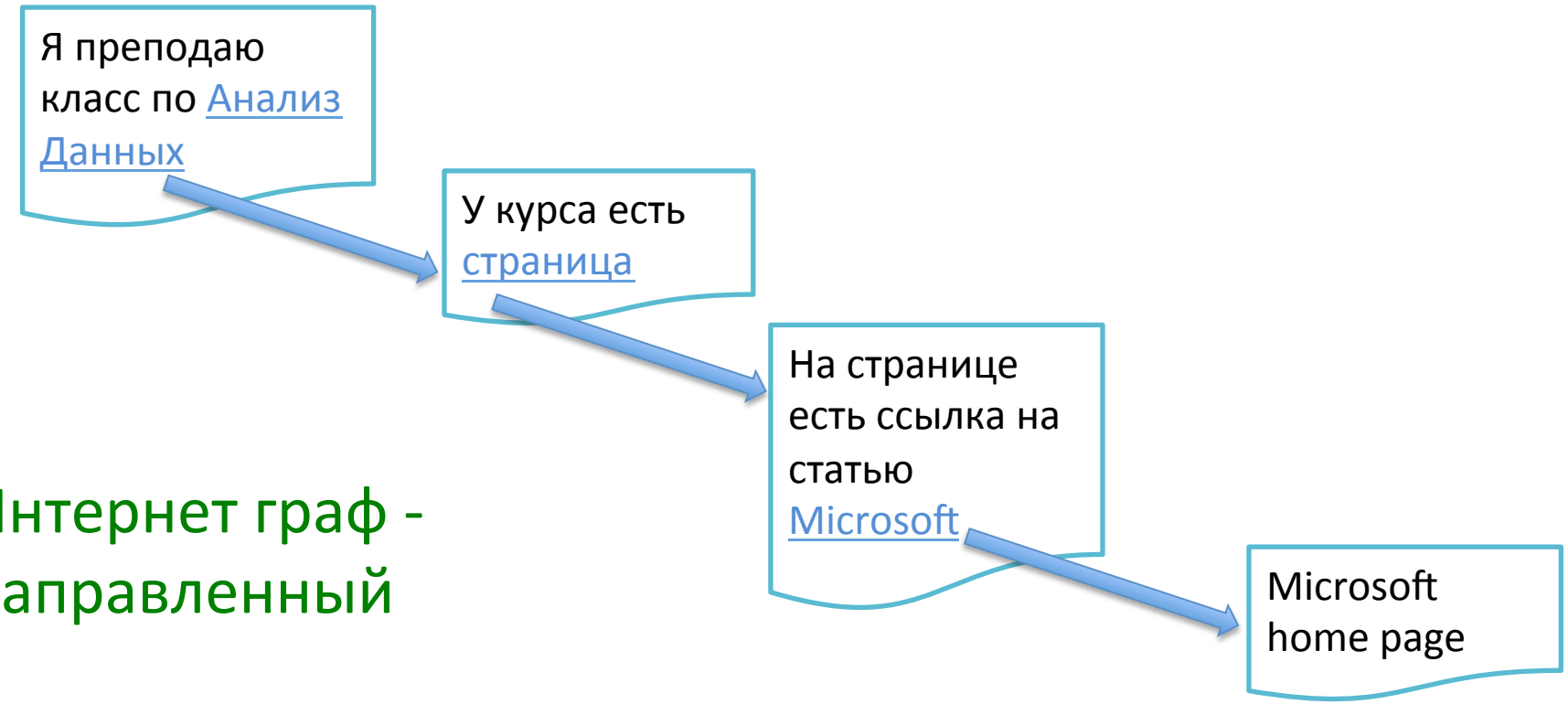
Интернет – это граф

- Какова структура Интернета?
- Как он организован?



Интернет – это граф

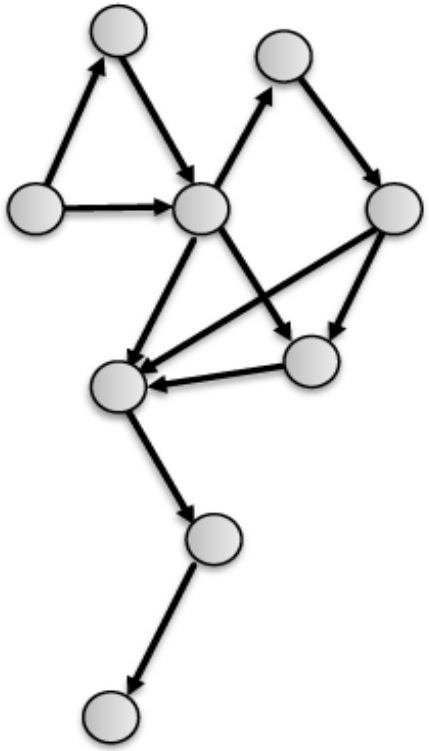
- Какова структура Интернета?
- Как он организован?



Интернет граф - направленный

Направленный граф

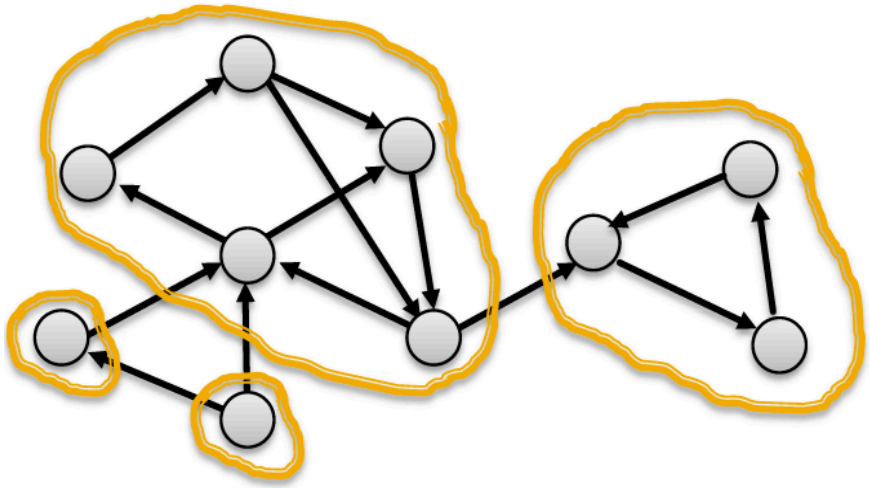
- Два типа направленных графов:
  - Направленный непериодический граф (ННГ):
    - не имеет цикла: если из ***u*** можно достигнуть ***v***, тогда из ***v*** нельзя достичь ***u***
  - Строго связанный граф:
    - Из любой вершины можно достичь любой вершины
- Любой направленный граф может быть описан с помощью этих двух типов графа



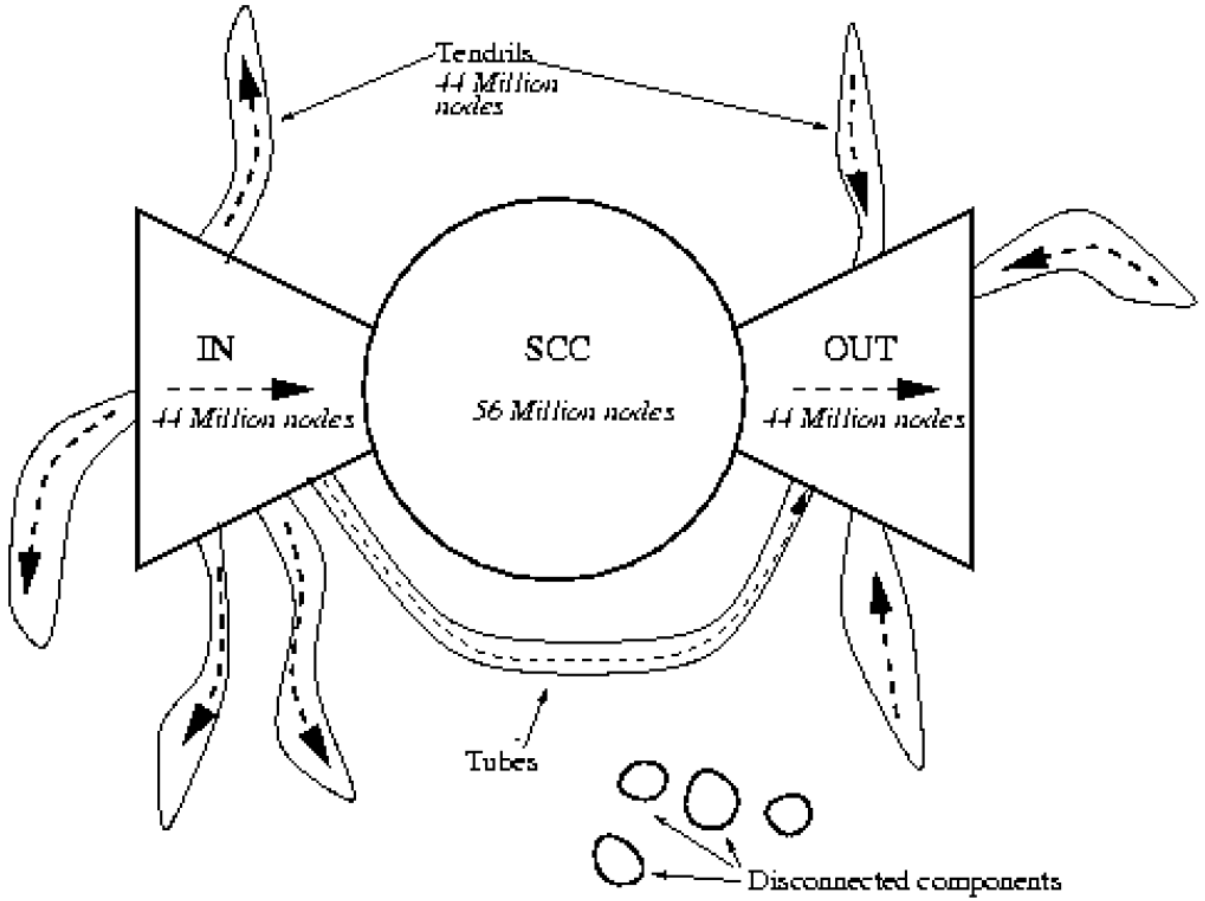
Строго связанный компонент

- Строго связанный компонент (ССК) - это набор узлов S:
  - из каждого узла S можно достигнуть другого узла
  - Не существует больше набора, содержащего S, которое обладает таким же свойством

- Любой направленный граф – это ННГ или ССК:
  - Каждый ССК – это супер-узел



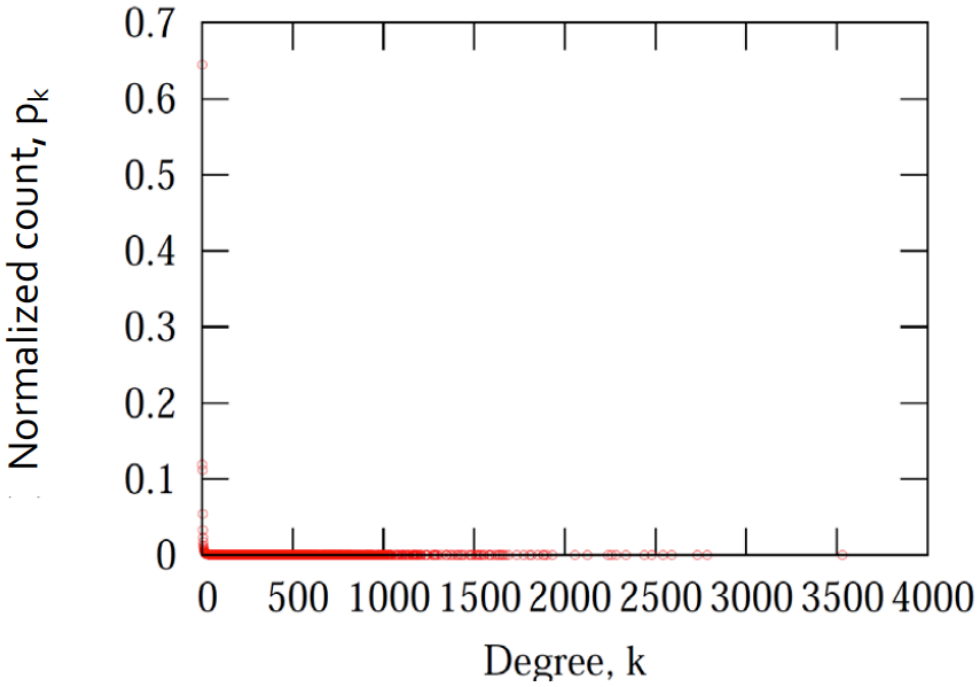
Бабочка структуры Интернета



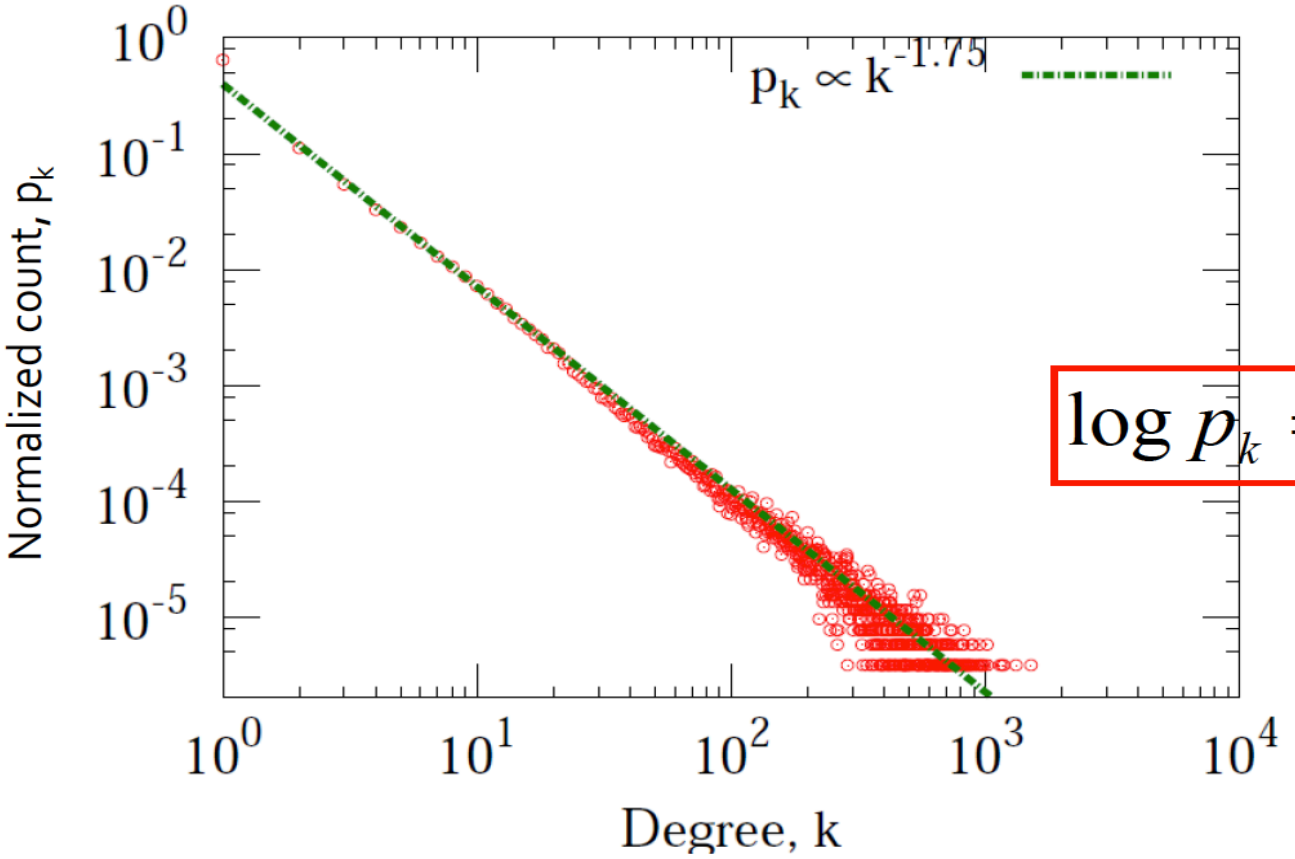
- 250 million webpages, 1.5 billion links [Altavista]

Степени значимости в реальных сетях

- Распределение Out-/In- уровней значимости:
  - p_k - доля узлов с k out-/in связей



Степени значимости в реальных сетях (2)



$$p_k = \beta k^{-\alpha}$$

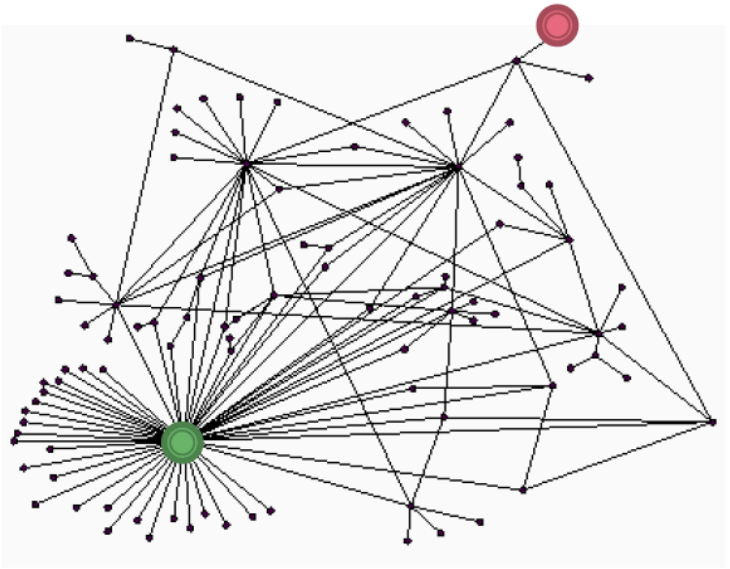
$$\log p_k = \log \beta - \alpha \log k$$

Ранжирование узлов в графе

- Веб-страницы не все одинаково «важны»
  - Например:

www.joe-schmoe.com vs. www.stanford.edu

- Так как существует большое разнообразие в связях Интернет-графа мы можем ранжировать страницы, используя структуру связей



План на сегодня

- Структура Интернета
- Page Rank

Алгоритмы анализа ссылок

- Существуют следующие методы для анализа ссылок для подсчета важности узлов в графе:
  - PageRank
  - Topic-Specific page Rank

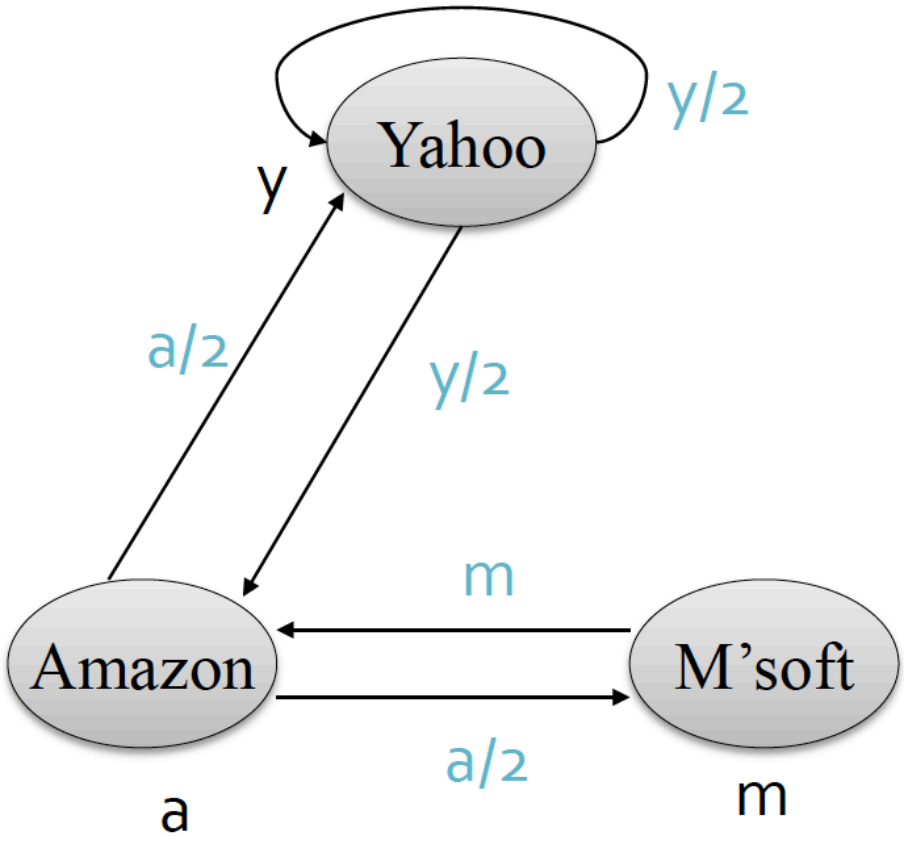
Рассмотрим ссылки как голоса

- Предположение: Страница более важна, если имеет много ссылок:
  - Входящие ссылки? Исходящие ссылки?
- Рассмотрим входящие связи как голоса:
  - www.stanford.edu имеет 23 400 входящих ссылок
  - www.joe-schmoe.com имеет 1 входящую ссылок
- Все ли входящие ссылки равнозначны?

Простое рекурсивное определение значимости

- Каждый голос ссылки пропорционален важности исходной страницы
- Если страница P с важностью x иммет n выходных ссылок, каждая ссылка получает x/n голосов
- Важность самой страницы P – это сумма всех голосов

Простая модель



$$y = y/2 + a/2$$
$$a = y/2 + m$$
$$m = a/2$$

Решение полученного уравнения

- 3 уравнения, 3 неизвестных, нет констант
  - Нет единственного решения
- Вводим дополнительные константы
  - y + a + m = 1
  - y= 2/5, a=2/5, m=1/5
- Метод Гаусса работает хорошо для небольших примеров, но нам нужен лучше метод для большого Интернет-графа

Марковский процесс

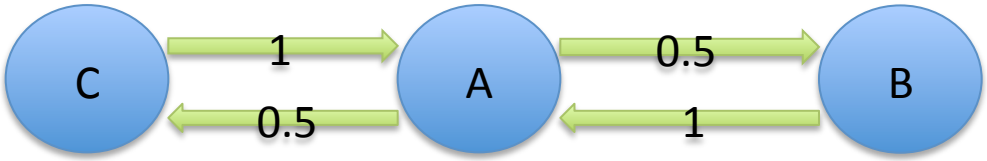
- *Марковская цепь* – это дискретный вероятностный процесс
- Марковский процесс состоит из N состояний => каждый сайт – это состояние
- Марковский процесс характеризуется матрицей вероятностей перехода P:

$$\forall i, j, P_{ij} \in [0,1]$$

$$\forall i, \sum_{j=1}^{i} P_{ij} = 1$$

- Вероятностная (стохастическая) матрица

Марковский процесс. Пример



	A	B	C
A	0	0.5	0.5
B	1	0	0
C	1	0	0

Матричная формулировка

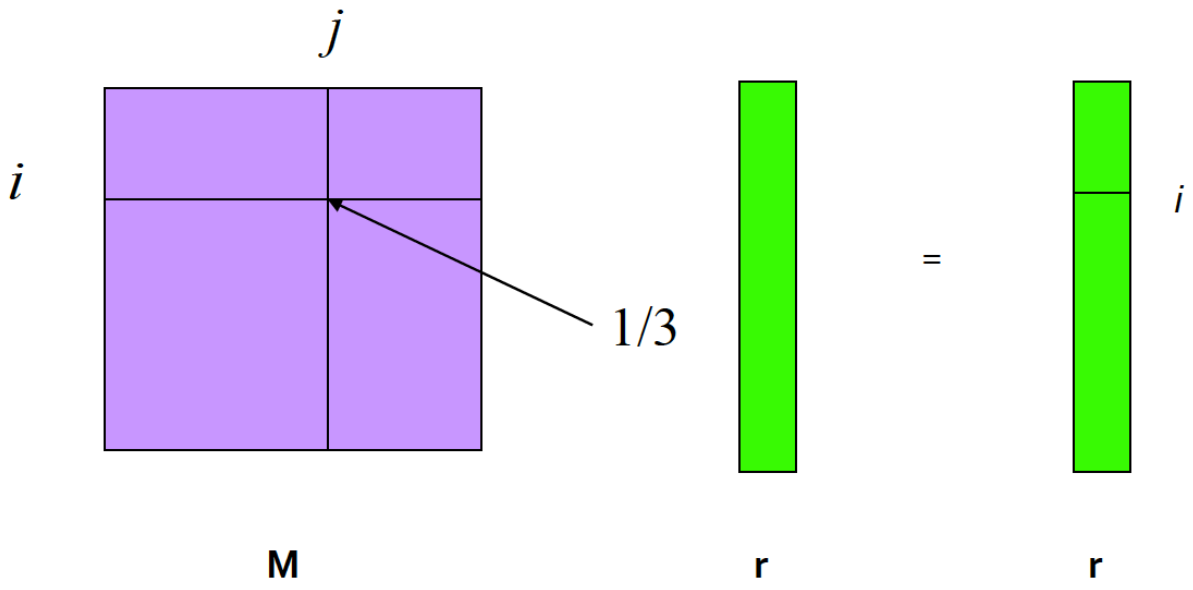
- В матрице **M** каждая строка и каждый столбец описывают одну веб-страницу
- Предположим у страницы j n выходных ссылок:
  - если *j -> i* => ***M****ij* = *1/n*
  - иначе ***M****ij* = *0*
- **M** – это вероятностная матрица

Матричная формулировка (2)

- Предположим *r* – это вектор, в котором каждая координата определяет страницу:
  - Координат *rj* – это оценка важности страницы *i (pageRank)*
  - Назовем этот вектор – вектором ранжирования
  - |r| = 1

Пример

- Предположим страница *j* связана с 3 страницами, включая *i*



Устойчивое состояние

Определение: Марковская цепь называется *эргодической*, если ∃ положительное число *To*, при котором все пары состояний *i, j* в Марковской цепи верно: если процесс начинается в состоянии *i* в момент времени 0, тогда для всех $t > To$ вероятность быть в состоянии *j* в момент времени *T* больше 0.

Устойчивое состояние (2)

Теорема: Для любой эргодической Марковской цепи $\exists$ уникальное устойчивое состояние, определяемое вектором $\pi(i)$, который является левым собственный вектором матрицы P:

$$\lim_{t\to\infty}\frac{\eta(i,t)}{t}=\pi(i)$$

где $\eta(i,t)$ - это число визитов в состояние *i за t* шагов

Напоминание: левый собственный вектор матрицы: $\overrightarrow{y}^{T} C = \lambda \overrightarrow{y}^{T}$

Устойчивое состояние(3)

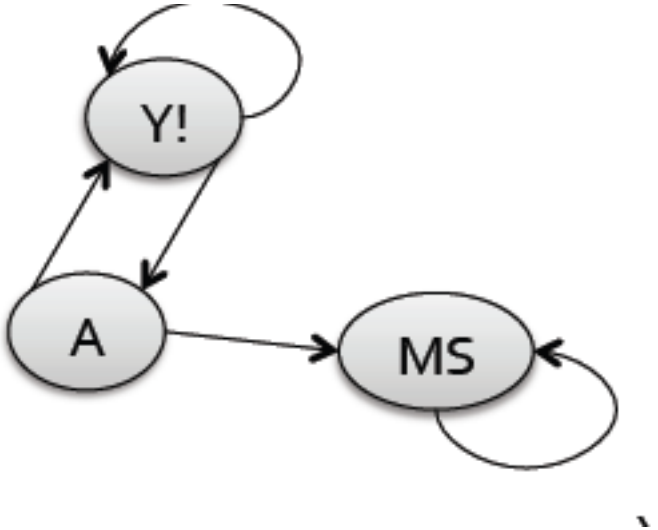
По теореме:

- $\vec{\pi}P = \lambda\vec{\pi}$ собственное число = 1 => $\vec{\pi}P = 1\vec{\pi}$

- **Power Iteration method**
-] N веб-страниц
- Инициализируем $\mathbf{r}^0 = [1/N,\ldots.,1/N]^T$
- Итерируем $r_{k+1} = r_k P^{k+1}$
- Останавливаемся $|r_{k+1} - r_k| < \varepsilon$

Существующие проблемы

- Некоторые страницы являются «dead-end» (нет исходящих ссылок)
- «Ловушка для паука» (все ссылки внутри одной группы)



Решение: teleports

- Google решение - > teleports
- В любой момент времени, обходчик имеет две возможности:
  - с вероятностью β, пойти по случайной ссылки
  - с вероятностью 1-β , перейти на случайную страницу
  - Обычно β принимает значения 0.9 или 0.8

Работа с dead-ends

- Предобработка графа с целью удаления dead-end
- Возможно необходимо несколько циклов
- Вычисление page rank на основе уменьшенного графа

Формализация матрицы

Пусть у нас *N* веб-страниц

- Рассмотрим страницу *j* с набором выходящих страниц *O(j)*
- *M(i,j) = 1/|O(j)|*, если *i->j*. Иначе *M(i,j)=0*
- Teleport эквивалентен:
  - Добавлению ссылки между *j* на любую другую страницу с вероятностью *(1- β)/N*
  - Уменьшению *1/ |Oj|* до β*/|Oj|*

Как посчитать pageRank?

- Строим матрицу **A N x N :**
  - **Aij = βMij + (1 -β)/N**
- Контролируем, что матрица A вероятностная (стохастическая)
- Page Rank вектор удовлетворяет :
  - r=A*r

Резюме

- Узнали немного про структуру интернета
- Узнали/вспомнили про ссылочное ранжирование