Машинное обучение

Лекция 15

Ранжирование

На прошлых лекциях

- Методы обучения с учителем: линейные модели, решающие деревья, случайные леса, …
- Дано: матрица «объекты-признаки» X и ответы y
- Найти: модель $a(x)$
- Модель должна выдавать прогнозы, близкие к истинным ответам

Задача Ранжирования

- Дано

  Матрица объекты-признаки $X = \{x_1, \dots x_n\}$

  Отношение x_j лучше x_i : $i \prec j$

- Найти

  Ранжирующую функцию a, которая восстанавливает правильное отношение порядка

$$i \prec j \Rightarrow a(x_i) < a(x_j)$$

Примеры Задач Ранжирования

- Ранжирование поисковой выдачи
- Ранжирование рекомендаций пользователям
- Ранжирование вариантов автоматического завершения запроса
- Ранжирование ответов в диалоговых системах

ПОИСК КАРТИНКИ ВИДЕО КАРТЫ МАРКЕТ НОВОСТИ ПЕРЕВОДЧИК ЕЩЁ

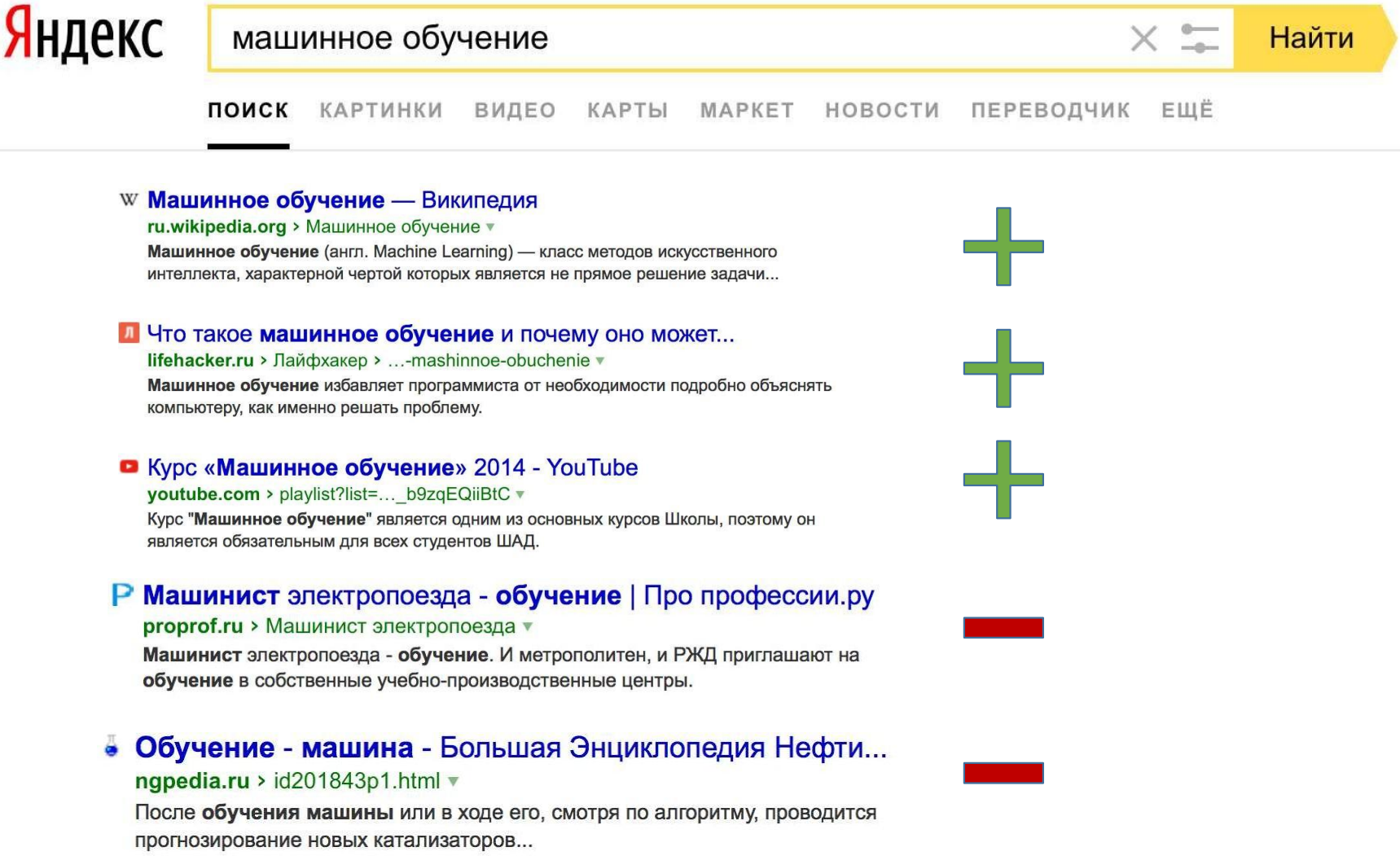
Машинное обучение — Википедия
ru.wikipedia.org › Машинное обучение
Машинное обучение (англ. Machine Learning) — класс методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи...

Что такое **машинное обучение** и почему оно может...
lifehacker.ru › Лайфхакер › …-mashinnoe-obuchenie
Машинное обучение избавляет программиста от необходимости подробно объяснять компьютеру, как именно решать проблему.

Курс «**Машинное обучение**» 2014 - YouTube
youtube.com › playlist?list=…_b9zqEQiiBtC
Курс "**Машинное обучение**" является одним из основных курсов Школы, поэтому он является обязательным для всех студентов ШАД.

Машинист электропоезда - **обучение** | Про профессии.ру
proprof.ru › Машинист электропоезда
Машинист электропоезда - **обучение**. И метрополитен, и РЖД приглашают на **обучение** в собственные учебно-производственные центры.

Обучение - машина - Большая Энциклопедия Нефти...
ngpedia.ru › id201843p1.html
После **обучения машины** или в ходе его, смотря по алгоритму, проводится прогнозирование новых катализаторов...


Яндекс
машинное обучение
Найти
ПОИСК
КАРТИНКИ
ВИДЕО
КАРТЫ
МАРКЕТ
НОВОСТИ
ПЕРЕВОДЧИК
ЕЩЁ
Машинное обучение — Википедия
ru.wikipedia.org › Машинное обучение
Машинное обучение (англ. Machine Learning) — класс методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи...
Что такое машинное обучение и почему оно может...
lifehacker.ru › Лайфхакер › …-mashinnoe-obuchenie
Машинное обучение избавляет программиста от необходимости подробно объяснять компьютеру, как именно решать проблему.
Курс «Машинное обучение» 2014 - YouTube
youtube.com › playlist?list=…_b9zqEQiiBtC
Курс "Машинное обучение" является одним из основных курсов Школы, поэтому он является обязательным для всех студентов ШАД.
Машинист электропоезда - обучение | Про профессии.ру
proprof.ru › Машинист электропоезда
Машинист электропоезда - обучение. И метрополитен, и РЖД приглашают на обучение в собственные учебно-производственные центры.
Обучение - машина - Большая Энциклопедия Нефти...
ngpedia.ru › id201843p1.html
После обучения машины или в ходе его, смотря по алгоритму, проводится прогнозирование новых катализаторов...

Ранжирование

- Дан набор запросов $\{q_1, \ldots, q_m\}$
- Дан набор документов $\{d_1, \ldots, d_n\}$
- Нужно для каждого запроса правильно упорядочить документы
- Что такое «правильно»?

Ранжирование

- Дан набор запросов $\{q_1, \dots, q_m\}$
- Дан набор документов $\{d_1, \dots, d_n\}$
- Рассматриваем пары «запрос-документ» (q, d)

Ранжирование

- Дан набор запросов $\{q_1, \dots, q_m\}$
- Дан набор документов $\{d_1, \dots, d_n\}$
- Рассматриваем пары «запрос-документ» (q, d)

- Известно, что для запроса q документ d_1 должен стоять раньше, чем d_2

  Обозначим это (q, d_1, d_2)

Ранжирование

- Дан набор запросов $\{q_1, \dots, q_m\}$
- Дан набор документов $\{d_1, \dots, d_n\}$
- Рассматриваем пары «запрос-документ» (q, d)

- Известно, что для запроса q документ d_1 должен стоять раньше, чем d_2

  Обозначим это (q, d_1, d_2)
- Обозначение R — множество троек, для которых известен такой порядок

Ранжирование

- Раньше: строим модель $a(x)$, которая приближает ответы
- Сейчас: строим модель $a(q, d)$, которая правильно упорядочивает документы для запросов

$$(q, d_1, d_2) \in R \Rightarrow a(q, d_1) > a(q, d_2)$$

Приме
р

- Для запроса q известны пары $(d_3, d_1), (d_3, d_2), (d_1, d_4)$
- Какие наборы прогнозов модели лучше?
  - $(3, 2, 4, 1)$
  - $(2, 3, 4, 1)$
  - $(3, 4, 2, 1)$
  - $(13, 10, 20, 7)$
  -

Приме
р

- Для запроса q известны пары $(d_3, d_1), (d_3, d_2), (d_1, d_4)$
- Какие наборы прогнозов модели лучше?
- $(\mathbf{3}, \mathbf{2}, \mathbf{4}, \mathbf{1})$
- $(\mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{1})$
- $(3, 4, 2, 1)$
- $(\mathbf{13}, \mathbf{10}, \mathbf{20}, \mathbf{7})$

- Важен порядок, а не абсолютные значения!

Метрики качества ранжирования

Задача Ранжирования

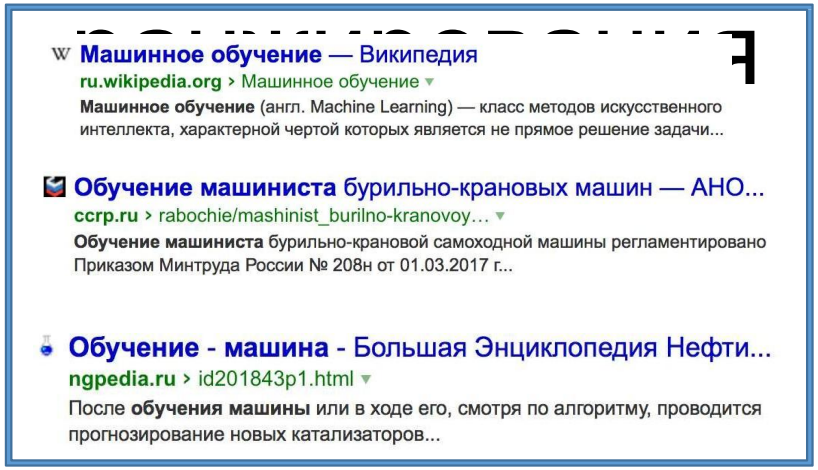
Матрица объекты-признаки: $X = \{x_1, \dots x_n\}$

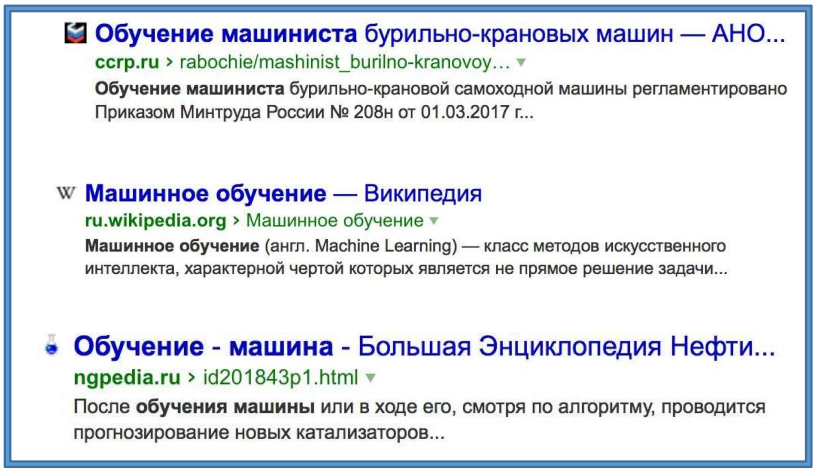
$$x_i = (q, d)$$

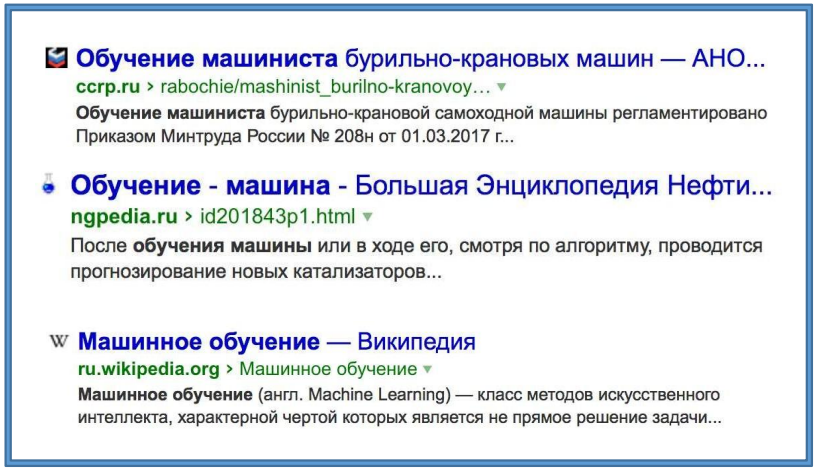
Ответы — числа y_i

Отношение x_j лучше x_i $y_j > y_i$

Качество







- Какое ранжирование лучше?
- Какое хуже всех?

Бинарные ответы $y_i \in \{0, 1\}$

- Документ либо соответствует запросу, или нет

- Можно применять стандартные метрики классификации:
  - Вычисляем метрику в рамках каждого запроса
  - Усредняем по всем запросам из выборки

Метрики классификации

$$\text{precision} = \frac{|\{\text{relevant}\} \cap \{\text{retrieved}\}|}{|\{\text{retrieved}\}|}$$

$$\text{recall} = \frac{|\{\text{relevant}\} \cap \{\text{retrieved}\}|}{|\{\text{relevant}\}|}$$

Метрики классификации

- Расссмотрим precision
- На самом деле нас интересуют документы, которые оказались вверху выдачи
- Поэтому рассматривают precision@k: метрика, вычисленная на k первых выдачах
- (!) Мы никак не учитываем порядок внутри первых k позиций

$$\mathrm{AP@}k(q) = \sum_{i=1}^{k} \frac{y_{(i)}}{\sum_{j=1}^{k} y_{(j)}} \mathrm{precision@}i(q),$$

Вещественная целевая переменная

- $(q_1, d_1), 1$
- $(q_1, d_2), 0.7$
- $(q_1, d_3), 0.1$
- $(q_2, d_1), 0.1$
- $(q_2, d_2), 1$
- Для q_1 должны получить ранжирование (d_1, d_2, d_3)
- Для q_2 должны получить ранжирование (d_2, d_1)

Метрики классификации

- Расссмотрим precision
- y_i - релевантность документа на i-й позиции
- Тогда можно посчитать Average Precision, где документ на первой позиции будет иметь больший вес

$$\mathrm{AP@}k(q) = \sum_{i=1}^{k} \frac{y_{(i)}}{\sum_{j=1}^{k} y_{(j)}} \mathrm{precision@}i(q),$$

DCG (Discounted cumulative gain)

$$\mathrm{DCG@}k(q) = \sum_{i=1}^{k} \frac{2^{y_i} - 1}{\log(i+1)}$$

- Вычисляется по первым k документам из выдачи для q запроса
- y_i — истинный ответ для документа на i-й позиции

- Чтобы получить итоговую оценку, DCG усредняется по всем запросам

Доля дефектных пар

$$\mathrm{DP@}k(q) = \frac{2}{k(k-1)} \sum_{i<j}^{k} [y_i < y_j]$$

- Число инверсий порядка среди k документо первых в

pFound

https://habr.com/ru/company/yandex/blog/197838/

pFound

- Вероятностная модель поведения пользователя
- При неуспехе с очередным документом выдачи пользователь разочаруется и уйдет с вероятностью P_{out}
- P_i — вероятность дойти до i-ого документа, y_i — вероятность того, что пользователь удовлетворится i-ым документом

$$P_1 = 1,$$
$$P_{i+1} = P_i (1 - y_i)(1 - P_{\text{out}})$$
$$\text{pFound@}k(q) = \sum_{i=1}^{k} P_i \, y_i$$

Разнообразие поисковой выдачи

- Неоднозначные запросы
- Пример: «ягуар»
  - Животное?
  - Марка автомобиля?
  - Танк? (немецкий или китайский?)
  - Напиток?

Разнообразие поисковой выдачи

- Неоднозначные запросы

- С точки зрения обычных метрик, весь топ выдачи нужно замостить одинаковыми релевантными документами
- Разнообразие позволяет собрать разнородную выдачу, чтобы удовлетворить в среднем всех

Wide pFound

- Предполагается, что пользователь, делая запрос, мог иметь в виду один из интентов $I = \{I_1, \ldots, I_m\}$
- Примеры интентов: автомобили, картинки, новости, животные, …
- Каждый интент имеет некоторую вероятность $p(I_i)$ и порождает собственное распределение релевантностей на документах

$$\text{wide pFound} = \sum_{i=1}^{m} p(I_i)\text{pFound}(I_i)$$

Wide pFound

- Как вычислить вероятности интентов?

Wide pFound

- Как вычислить вероятности интентов?

- Интент пользователя определяется по продолжениям введенного запроса
- Продолжения классифицируются по различным тематикам
- Тематики являются интентами
- Вероятности определяются по частоте соответствующих продолжений запросов

Признаки в задачах ранжирования

Типы признаков

- Запросные
  - Популярность запроса
  - Тип запроса (навигационный, товарный и т.д.)
- Статические — зависят только от документа
  - Популярность документа
  - Тематика
  - Распределение слов
- Динамические — зависят от документа и от запроса
  - Расстояния между запросом и документом

Признаки ранжирования Google

- https://backlinko.com/google-ranking-factors

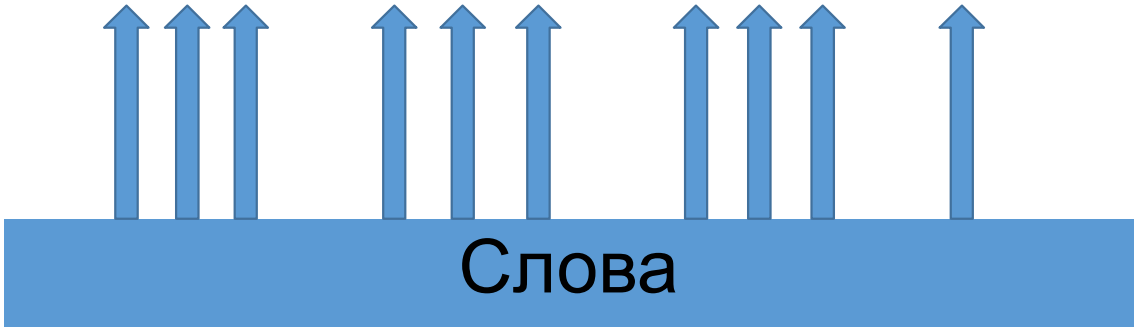
Мешок слов

- $v(\text{большое}) = (1, 0, 0, 0, \ldots, 0)$
- $v(\text{спасибо}) = (0, 1, 0, 0, \ldots, 0)$
- $v(\text{минус}) = (0, 0, 1, 0, \ldots, 0)$
- $v(\text{зарубежный}) = (0, 0, 0, 1, \ldots, 0)$
- …
- $v(\text{инквизиция}) = (0, 0, 0, 0, \ldots, 1)$

Мешок слов

- Текст — это вектор x, содержащий счётчики слов

$$(0, 2, 0, \ldots, 1, 10, 0, \ldots, 0, 1, 0, \ldots, 1)$$



Косинусное расстояние

- Пусть $\vec{q}$ — вектор запроса, $\vec{d}$ — вектор документа
- Мера сходства:

$$s(\vec{q}, \vec{d}) = \frac{\sum_{i=1}^{n} q_i d_i}{\|\vec{q}\| \|\vec{d}\|}$$

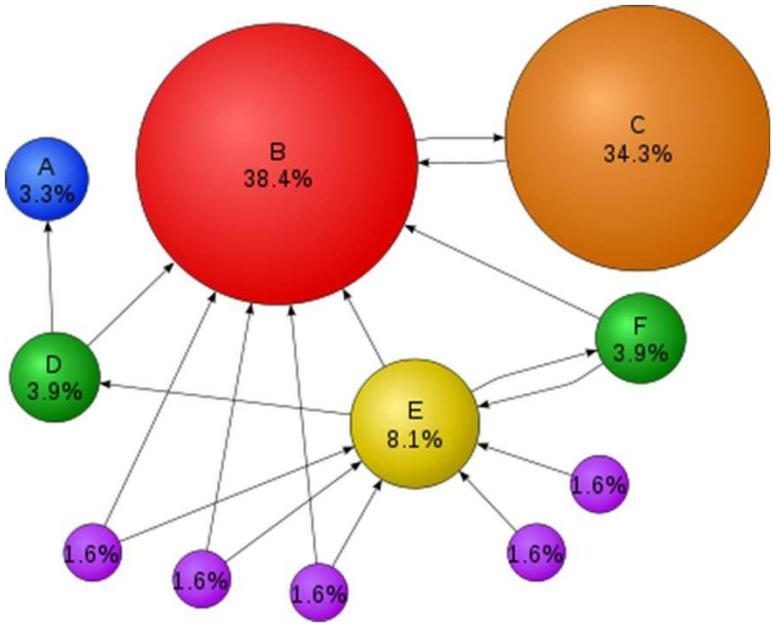
- Чем больше, тем сильнее тексты похожи по долям слов

Продвинутое расстояние: BM25

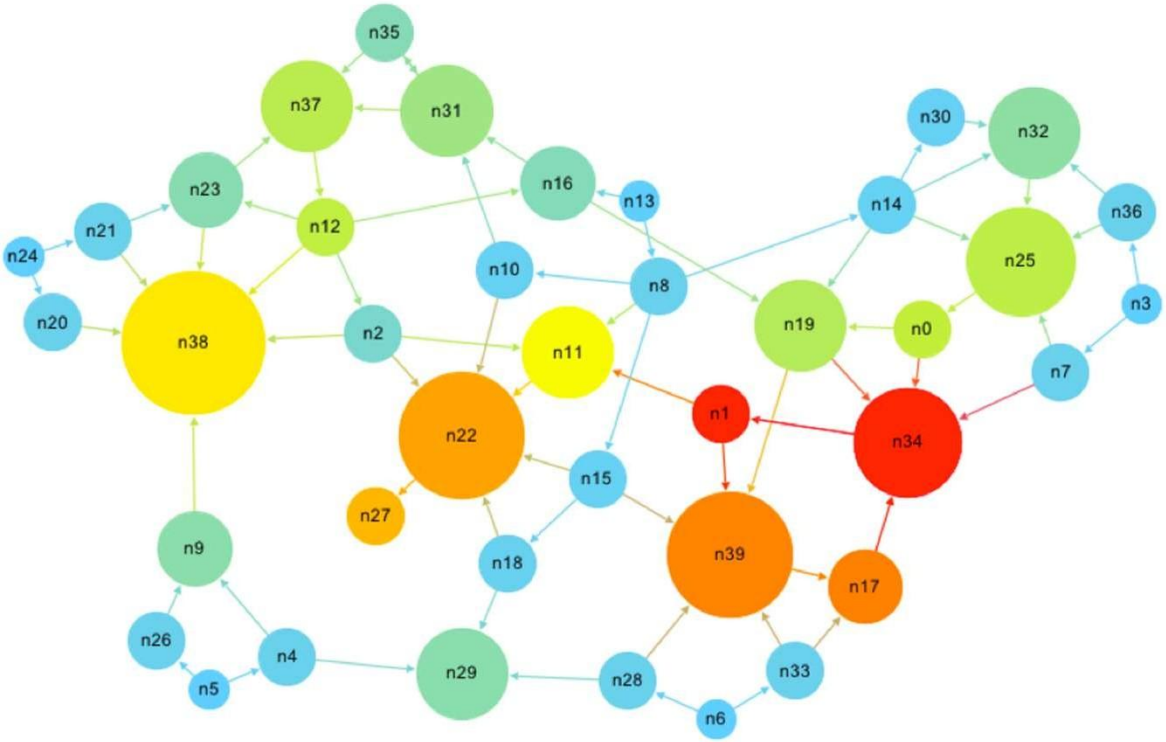
$$\mathrm{BM25}(q,d) = \sum_{i=1}^{n} \mathrm{IDF}(q_i) \frac{\mathrm{tf}(q_i,d)(k_1+1)}{\mathrm{tf}(q_i,d) + k_1\left(1 - b + b\frac{|D|}{\bar{n}_d}\right)}$$

PageRank

- Документы в сети ссылаются друг на друга
- Если документ A ссылается на документ B, то он «голосует» за B
- Чем меньше голосов отдаёт A, тем сильнее его голос
- Документ B важен, если за него отдано много сильных голосов



PageRank


n35
n37
n31
n30
n32
n23
n16
n13
n14
n36
n21
n12
n25
n24
n10
n8
n3
n20
n38
n2
n11
n19
n0
n7
n1
n34
n22
n15
n27
n9
n18
n39
n17
n26
n4
n29
n28
n33
n5
n6

PageRank

- Пусть пользователь бродит по сети
- Стартует из случайного документа
- С вероятностью $(1-\delta)$ переходит по одной из ссылок с равными вероятностями
- С вероятностью δ переходит на случайный документ из всей сети
- PageRank — вероятность при таком случайном блуждании попасть в данный документ

PageRank

- PageRank страницы u зависит от PageRank страниц v из множества B_u (страниц, которые ссылаются на u), поделенного на число исходящих ссылок $L(v)$ из страницы v:

$$\mathrm{PR}(u) = \sum_{v \in B_u} \frac{\mathrm{PR}(v)}{L(v)}$$

PageRank

- Учтем, что пользователь может остановиться в какой-то момент
- Установим damping factor (фактор затухания) – обычно $d \approx 0.85$
- N – число рассматриваемых страниц

$$\mathrm{PR}(u) = \frac{1-d}{N} + d \sum_{v \in B_u} \frac{\mathrm{PR}(v)}{L(v)}$$

Методы ранжирования

Поточечный (pointwise) подход

- Обучим модель $a(q,d)$, чтобы она как можно точнее приближала y_i ответы
- Например, линейная регрессия: $\sum_{(q,d,y)\in R} (\langle w, x(q,d)\rangle - y_i)^2 \to \min_w$

- $x(q,d)$ — признаки для пары «запрос-документ»

Поточечный (pointwise) подход

- Простой в реализации
- Можно использовать любую из известных моделей (линейные, деревья, случайные леса, нейронные сети…)
- Восстанавливает точные значения y_i, хотя нас интересует порядок

Попарный (pairwise) подход

- В ранжировании требуется правильно располагать пары документов — формализуем это

$$\sum_{(q,d_i,d_j)\in R} \left[a(q,d_i) - a\left(q,d_j\right) < 0\right]$$

- Штрафуем, если второй документ из пары оказался раньше

Попарный (pairwise) подход

- Получили разрывный функционал — сложно оптимизировать
- Перейдём к гладкой верхней оценке (как в линейных классификаторах):

$$\sum_{(q,d_i,d_j)\in R} [a(q,x_i) - a(q,x_j) < 0] \leq \sum_{(q,d_i,d_j)\in R} L\left(a(q,x_i) - a(q,x_j)\right)$$

RankNet

$$\sum_{(q,d_i,d_j)\in R} \left[a(q,x_i) - a\left(q,x_j\right) < 0\right] \leq \sum_{(q,d_i,d_j)\in R} L\left(a(q,x_i) - a\left(q,x_j\right)\right)$$

- Выберем логистическую функцию:

$$L(z) = \log(1 + e^{-z})$$

Попарный (pairwise) подход

- Сложнее поточечного (больше слагаемых в функционале)
- Обычно даёт качество выше, чем поточечный
- Реализаци $\mathrm{SVM}^{\mathrm{light}}$, xgboost (rank:pairwise) и:

Резюме

- Ранжирование — задача сортировки документов по релевантности
- Метрика должна учитывать позиции, а не абсолютные значения прогнозов — например, DCG
- Поточечный и попарный подходы
- Отдельная задача — разработка признаков