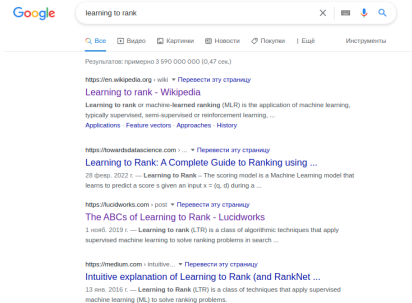


Ранжирование (Learning to Rank)

Виктор Китов

v.v.kitov@yandex.ru

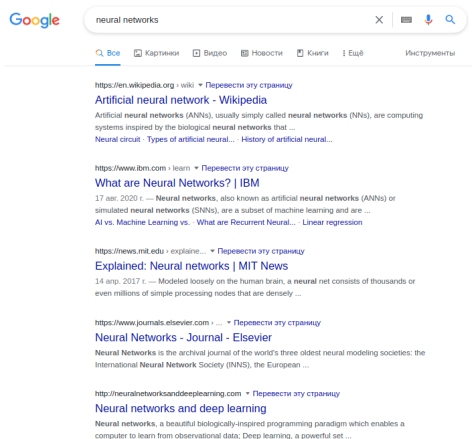


Содержание

- 1 Постановка задачи
- 2 Формирование вектора признаков
- 3 Поточечные (pointwise) методы
- 4 Попарные (pairwise) методы
- 5 Списочные (listwise) методы
- 6 Меры качества ранжирования

Примеры ранжирования

Ранжирование веб-страниц по запросу:



The screenshot shows a Google search for "neural networks". The search bar is at the top with the text "neural networks" and a magnifying glass icon. Below the search bar are several tabs: "Все" (All), "Картинки" (Images), "Видео" (Videos), "Новости" (News), "Книги" (Books), "Ещё" (More), and "Инструменты" (Tools). The search results are listed below, each with a URL, a title, and a brief description.

Google

neural networks

Все Картинки Видео Новости Книги Ещё Инструменты

<https://en.wikipedia.org> > wiki > Перевести эту страницу
Artificial neural network - Wikipedia
Artificial neural networks (ANNs), usually simply called neural networks (NNs), are computing systems inspired by the biological neural networks that ...
Neural circuit - Types of artificial neural... - History of artificial neural...

<https://www.ibm.com> > learn > Перевести эту страницу
What are Neural Networks? | IBM
17 авг. 2020 г. — Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are ...
AI vs. Machine Learning vs. - What are Recurrent Neural... - Linear regression

<https://news.mit.edu> > explaine... > Перевести эту страницу
Explained: Neural networks | MIT News
14 апр. 2017 г. — Modeled loosely on the human brain, a neural net consists of thousands or even millions of simple processing nodes that are densely ...

<https://www.journals.elsevier.com> > ... > Перевести эту страницу
Neural Networks - Journal - Elsevier
Neural Networks is the archival journal of the world's three oldest neural modeling societies: the International Neural Network Society (INNS), the European ...

<http://neuralnetworksanddeeplearning.com> > Перевести эту страницу
Neural networks and deep learning
Neural networks, a beautiful biologically-inspired programming paradigm which enables a computer to learn from observational data; Deep learning, a powerful set ...

Примеры ранжирования

Ранжирование ответов на вопрос:

StackExchange

mathoverflow

Home
PUBLIC
Questions
Tags
Users
Unanswered

Search Results

Advanced Search Tips [Ask Question](#)

Results for compute svd decomposition
Search options not deleted

28 results

Relevance	Newest	Score	Active
8 votes	Efficient SVD of a matrix without some of the columns		
1 answer	I have a matrix $A \in \mathbb{R}^{p \times q}$ of rank r and its SVD decomposition , i.e.,		
1k views	$A = U S V^T$,		
where $U \in \mathbb{R}^{p \times p}$ and $V \in \mathbb{R}^{q \times q}$ are orthonormal ... I would like to compute SVD for...			
matrices computer-science matrix-analysis computer-algebra			
manan asked Nov 7, 2015 at 19:32			
1 vote	Complexity of singular value decomposition using matrix multiplication oracles		
0 answers	I have an oracle that can compute Ax or $A^T y$ for any $x \in \mathbb{R}^n$, $y \in \mathbb{R}^k$ How many oracle calls do I need (asymptotically) to calculate the "economy" SVD decomposition of A , namely...		
17 views	linear-algebra numerical-analysis numerical-linear-algebra		
AspiringMat asked Apr 21 at 4:43			
4 votes	Making MATLAB svd robust to transpose operation		
3 answers	I'm playing with MATLAB's svd function to compute the svd of $\begin{bmatrix} 1 & 4 & 7 & 10 & 2 & 5 & 8 & 11 & 3 & 6 & 9 & 12 \end{bmatrix}$ When I type $[U1, \sim, \sim] = \text{svd}(X)$, I get $U1 = -0.5045 \ 0.7608 \ \dots \ 0.4082 \ -0.5745 \ 0.0571 \ -0.8165 \ -0.6445 \ -0.646 \ \dots$		
3k views	linear-algebra numerical-analysis		
Ismael An asked Oct 11, 2010 at 7:42			

Примеры ранжирования


Ранжирование товаров по запросу:

amazon Deliver to Russian Federation All wireless headphones

All Today's Deals Customer Service Registry Gift Cards Sell

1-16 of over 2,000 results for "wireless headphones"

RESULTS




Featured from our brands

Tiksounds Wireless Earbuds, Bluetooth Headphones with Microphone, IPX7 Waterproof, 35H Playtime, High-Fidelity Stereo Earphones for Sports and Work


★★★★☆ - 11,457

Exclusive to Amazon



Beats Solo3 Wireless On-Ear Headphones - Apple W1 Headphone Chip, Class 1 Bluetooth, 40 Hours of Listening Time, Built-in Microphone - Black (Latest Model)

★★★★☆ - 41,757



Powerbeats Pro Wireless Earbuds - Apple H1 Headphone Chip, Class 1 Bluetooth Headphones, 9 Hours of Listening Time, Sweat Resistant, Built-in Microphone - Black

★★★★☆ - 72,353

Постановка задачи

- u - пользователь
- q - поисковый запрос
- d - ранжируемая сущность (например документ поисковой выдачи)
 - др. примеры: веб-страницы, ответы на вопрос, товары, ...
- Изучаемый объект - тройка:

$$x = (u, d, q)$$

- Задача: построить $f(x)$, выдающий
 - большое значение для релевантных объектов
 - малое значение - для нерелевантных

Содержание

- 1 Постановка задачи
- 2 Формирование вектора признаков
 - PageRank
- 3 Поточечные (pointwise) методы
- 4 Попарные (pairwise) методы
- 5 Списочные (listwise) методы
- 6 Меры качества ранжирования

Формирование вектора признаков

- **Формирование вектора признаков:**

$$(u, q, d) \rightarrow (f_1(u), f_2(q), f_3(d), f_4(q, d))$$

Более сложный случай: рассмотреть взаимодействие пользователя с запросом и документом.

- **Примеры пользовательских признаков:**
 - пол, возраст
 - интересы, предыдущий запрос
 - вариативность кликов
 - средняя глубина просмотра результатов

Формирование вектора признаков

- **Примеры запросных признаков:**
 - длина, счетчики слов, doc2vec
 - популярность запроса
 - тема запроса (среди заданных или определяемых тематич. моделированием)
 - тип запроса:
 - навигационный (введен веб-адрес),
 - однозначный (например "википедия")
 - товарный ("купить беспроводные наушники")
 - вопросный ("курс доллара")
 - информационный ("группа Cranberries")
 - ...

Формирование вектора признаков

- **Примеры документных признаков:**
 - время создания
 - длина документа, заголовка, #изображений, #ссылок
 - счетчики слов, doc2vec
 - тема документа (среди заданных или определяемых тематич. моделированием)
 - популярность документа (#кликов, #ссылок на него, PageRank)

- 2 Формирование вектора признаков
 - PageRank

PageRank - идея

- PageRank - расчет рейтинга веб-страниц.
- Идея: документ d важен, если на него ссылаются много др. важных документов.
 - А важен, т.к. на него много ссылок
 - В важен, т.к. на него ссылается А



PageRank

- $N = \#$ документов
- $D_{in}(d)$ -документы, ссылающиеся на d
- $D_{out}(d')$ -документы, на которые ссылается d'
- Рассмотрим пользователя, который
 - с $p = \varepsilon$ (параметр=0.15) переходит на случайный документ
 - с $p = 1 - \varepsilon$ переходит по ссылке в текущем документе
- π_d - вероятность оказаться в d после бесконечно долгого блуждания.
- Наивный подход: стартуя с $\pi = [1/N, \dots, 1/N]$, пересчитываем до сходимости

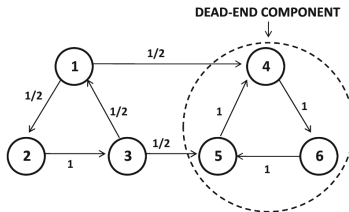
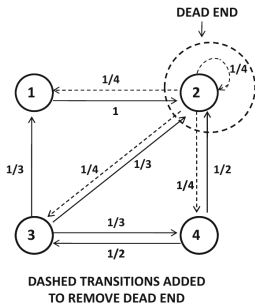
$$\pi_d := \sum_{d' \in D_{in}(d)} \frac{\pi_{d'}}{|D_{out}(d')|}$$

PageRank - важность случайного перехода¹

Чтобы не застревать в документе без ссылок, **добавляем искусственные ссылки** (пунктиром) во все документы.

Чтобы не застревать в замкнутом сообществе: с $p = \varepsilon$ переходим в случайный документ:

$$\pi_d = \frac{\varepsilon}{N} + (1 - \varepsilon) \sum_{d' \in D_{in}(d)} \frac{\pi_{d'}}{|D_{out}(d')|}$$



PageRank - расчёт

- $e = [1, \dots, 1] \in \mathbb{R}^N$
- $P \in \mathbb{R}^{N \times N}$ - вероятности переходов
 - для документов без внешних ссылок - равномерные ссылки на все документы

$$p_{ij} = \frac{\mathbb{I}\{i \text{ ссылается на } j\}}{\# \text{ссылок из } i}$$

$$\text{Для } P^T : p_{ji}^T = \frac{\mathbb{I}\{i \text{ ссылается на } j\}}{\# \text{ссылок из } i}$$

- Стартуя с $\pi = \frac{1}{N}e$, пересчитываем до сходимости²:

$$\pi := \varepsilon \frac{1}{N}e + (1 - \varepsilon) P^T \pi$$

²Нужно ли перенормировать π после каждой итерации для $\sum_d \pi_d = 1$?

Тематический PageRank

Тематический PageRank: насколько документ важен в теме?

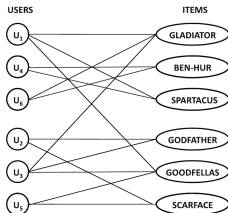
- например, автомобили.
 - англ. topic-sensitive, personalized PageRank
- 1 Выделяем документы, релевантные теме t (N_t штук)
 - обозначим $\mathbf{e}_t = [\mathbb{I}\{d_1 \in t\}, \dots, \mathbb{I}\{d_N \in t\}]$
 - 2 Инициализируем $\pi = \mathbf{e}_t$
 - 3 Пересчитываем до сходимости

$$\pi := \varepsilon \frac{1}{N} \mathbf{e}_t + (1 - \varepsilon) P^T \pi$$

Тематический PageRank

Комментарии:

- Для каждой темы t_1, t_2, \dots - свой расчёт
- для комбинации тем - лин. комбинация тематических PageRank.
- частный случай (personalized PageRank): $e_{ti} = \mathbb{I}\{i = d\}$
 - посчитается близость документов к d на графе.
 - полезно в рекомендательных системах (близость i к u)



Признаки взаимодействия для ранжирования

- **Примеры признаков взаимодействия $f_4(d, q)$**
(самый информативный тип признаков)

- число общих слов, общих фраз
- BM-25³ и др. вариации следующего признака:

$$f(q, d) = \sum_{w \in q} IDF(w | \{d_i\}_i) \cdot TF(w | d)$$

- $\text{cos-sim}(\text{embedding}(d), \text{embedding}(q))$
 - $\text{embedding}(d)$ - можно заранее посчитать
 - embedding можно обучить, используя triplet или contrastive loss => сиамская сеть
 - быстрый и достаточно точный для первичного отсева нерелевантных документов

³https://en.wikipedia.org/wiki/Okapi_BM25

Ранжирование в реальных системах

- ❶ Пользователь формирует запрос.
- ❷ Грубый отбор кандидатов ($\sim 10^3$)
 - например, $\text{cos-sim}(\text{embedding}(d), \text{embedding}(q)) \geq \text{threshold}$
- ❸ Ранжирование
- ❹ Коррекция результатов
 - обеспечение разнообразия результатов
 - фильтрация спама
- ❺ Выдача

Сбор обучающей выборки

- Отклики $x \rightarrow y$ можем собирать:
 - из пользовательских данных
 - пользователь ввел запрос и кликнул/не кликнул
 - усредняя, получим вероятность клика при запросе
 - важно учитывать время на веб-странице, был ли впоследствии возобновлен поиск
 - из ручной разметки ассессорами по шкале
 - релевантен / не релевантен
 - степень релевантности (более информативно)
- Ручная разметка более точная, т.к. пользователь
 - при клике видит лишь сниппет
 - может увлечься документом, хотя он нерелевантный
 - не учитываются релевантные документы в конце выдачи

Содержание

- 1 Постановка задачи
- 2 Формирование вектора признаков
- 3 Поточечные (pointwise) методы**
- 4 Попарные (pairwise) методы
- 5 Списочные (listwise) методы
- 6 Меры качества ранжирования

Поточечные (pointwise) методы

Поточечные (pointwise) методы ранжирования:

- $y \in \{1, 0\}$ (релевантен/не релевантен)
 - бинарная классификация
 - упорядочиваем выдачу по $p(y = +1|x)$
 - регрессия на $+1, 0$.
- $y \in \mathbb{R}$ - степень релевантности. Тогда методы:
 - многоклассовая классификация
 - не учитывается порядок классов (нерел. \prec рел. \prec сильно рел.)
 - регрессия
 - оценки релевантности - субъективны и произвольны
 - пользователю важен лишь порядок $x_1 \succ x_2 \succ x_3 \succ \dots$
 - т.е. чтобы $f(x_1) > f(x_2) > f(x_3) > \dots$, сами значения $f(x)$ не важны.

Порядковая регрессия (ordinal regression)

- Как по релевантностям предсказывать релевантности, учитывая упорядоченность классов?

Порядковая регрессия (ordinal regression)

- Как по релевантностям предсказывать релевантности, учитывая упорядоченность классов?
- **Порядковая регрессия (ordinal regression)** - метод классификации для упорядоченных классов.
 - степень релевантности, оценка на экзамене, уровень удовлетворенности услугой, ...
 - $x \rightarrow z(x) \in \mathbb{R} \rightarrow y \in \{1, 2, \dots, K\}$, используя разбиение шкалы z на интервалы

Порядковая регрессия (ordinal regression)

Предполагаем соответствие между вещественной $z(x)$ и порядковой $y(x)$, а также вид зависимости $z(x)$:

$$y = \begin{cases} 1 & z \leq \theta_1 \\ 2 & \theta_1 < z \leq \theta_2 \\ 3 & \theta_2 < z \leq \theta_3 \\ \dots & \dots \\ K & \theta_{K-1} < z \end{cases}$$

$z = w^T x + \varepsilon$, $\varepsilon \sim F(\cdot)$ ф-ция распределения

$$\begin{aligned} P(y = k|x) &= P(\theta_{k-1} < z \leq \theta_k|x) = P(\theta_{k-1} < w^T x + \varepsilon \leq \theta_k) \\ &= F(\theta_k - w^T x) - F(\theta_{k-1} - w^T x) \end{aligned}$$

$$\prod_{n=1}^N P(y_n|x_n) \rightarrow \max_{w, \theta} \text{ - находим веса и пороги}$$

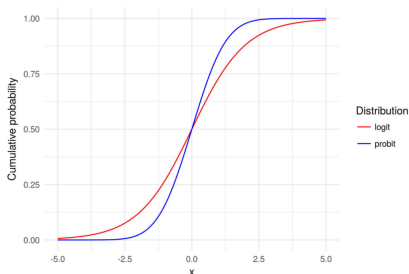
Порядковая регрессия (ordinal regression)

Популярные $F(\cdot)$:

$$F(u) = \sigma(u) = \frac{1}{1 + e^{-u}} \text{ - ordinal logit model}$$

$$F(u) = P(\eta \leq u), \eta \sim \mathcal{N}(0, 1) \text{ - ordinal probit model}$$

Logit и Probit функции распределения:



Обобщение порядковой регрессии через ML^4

- Обобщение: $z = f_w(x), \quad y = k \iff \theta_{k-1} < f_w(x) \leq \theta_k$.
- Обучение по ближайшим порогам:

$$\sum_{n=1}^N S(\theta_{y_n} - f_w(x_n)) - S(\theta_{y_n-1} - f_w(x_n)) \rightarrow \max_{w, \theta}, \quad S(\cdot) \uparrow$$

- Обучение по всем порогам:

$$\sum_{n=1}^N \sum_{k=1}^{K-1} S((\theta_k - f_w(x_n)) \operatorname{sign}(k - y_n)) \rightarrow \max_{w, \theta}, \quad S(\cdot) \uparrow$$

$$\operatorname{sign}(u) = \begin{cases} +1 & u \geq 0 \\ -1 & u < 0 \end{cases}$$

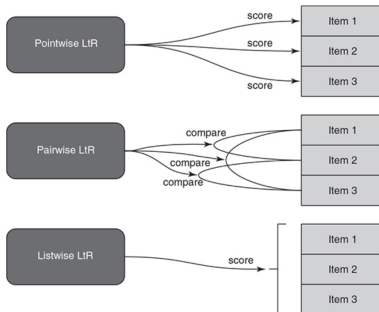
- Можно с регуляризацией.

⁴ Loss functions for preference levels.

Методы ранжирования

Методы ранжирования:

- поточечные
- попарные (самые популярные)
- списочные (в теории - самые точные)



Содержание

- 1 Постановка задачи
- 2 Формирование вектора признаков
- 3 Поточечные (pointwise) методы
- 4 Попарные (pairwise) методы**
- 5 Списочные (listwise) методы
- 6 Меры качества ранжирования

Попарные (pairwise) методы

- Поточечные методы прогнозируют $x \rightarrow y$.
 - неоднозначность назначения релевантностей
 - важен корректный порядок, а не точные прогнозы y
 $x_i \succ x_j \succ \dots$ может достигаться разными $y_i \succ y_j \succ \dots$
- Попарные (pairwise) методы решают напрямую задачу упорядочивания.

Попарные (pairwise) методы

$$j \succ i \iff f_w(x_i) < f_w(x_j), \quad f_w(x) = w^T x = w^T g(d, q) \text{ (например)}$$

$$\sum_{j \succ i} \mathbb{I}[f_w(x_i) \geq f_w(x_j)] \leq \sum_{j \succ i} \mathcal{L}(f_w(x_j) - f_w(x_i)) \rightarrow \min_w$$

Примеры $\mathcal{L}(u) = \mathcal{L}(f_w(x_j) - f_w(x_i))$, $\mathcal{L}(\cdot) \downarrow$:

$\mathcal{L}(u) = e^{-u}$	RankBoost
$\mathcal{L}(u) = \log_2(1 + e^{-u})$	RankNet
$\mathcal{L}(u) = \max\{1 - u; 0\}$	RankSVM

RankSVM⁵

$$\frac{1}{2} \|w\|_2^2 + C \sum_{i \prec j} \max \{1 - (f_w(x_j) - f_w(x_i)); 0\} \rightarrow \min_w$$

соответствует задаче квадратичной оптимизации для $f_w(x) = w^T x$:

$$\begin{cases} \frac{1}{2} \|w\|_2^2 + C \sum_{j \succ i} \xi_{ij} \rightarrow \min_{w, \xi} \\ w^T (x_j - x_i) \geq 1 - \xi_{ij} & i \prec j \\ \xi_{ij} \geq 0 & i \prec j \end{cases}$$

и допускает ядерное обобщение (kernel trick).

⁵Ching-Pei Lee and Chih-Jen Lin. Large-scale Linear RankSVM

RankNet

$$\sum_{j \succ i} \log_2 \left(1 + e^{-\alpha(w^T x_j - w^T x_i)} \right) \rightarrow \min_w$$

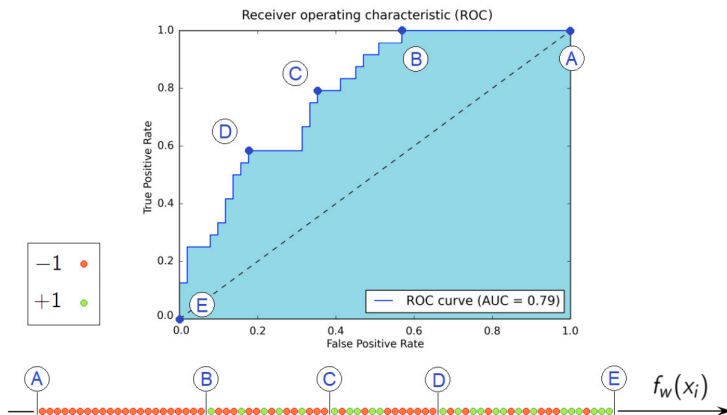
Шаг SGD по случайной паре $x_i \prec x_j$:

$$w := w - \varepsilon \frac{e^{-\alpha(w^T x_j - w^T x_i)}}{1 + e^{-\alpha(w^T x_j - w^T x_i)}} (-\alpha) (x_j - x_i)$$

$$w := w + \varepsilon \alpha \frac{1}{1 + e^{+\alpha(w^T x_j - w^T x_i)}} (x_j - x_i)$$

Интуиция: чем сильнее $w^T x_i > w^T x_j$, тем сильнее сдвигаем w в сторону x_j , чтобы $\uparrow w^T x_j$.

Напоминание AUC



Оптимизация AUC

- Эквив. определение - доля верно упорядоченных пар:

$$AUC = \frac{\sum_{(i,j): y_i=-1, y_j=1} \mathbb{I}[f_w(x_j) > f_w(x_i)]}{\#[i : y_i = -1] \#[j : y_j = 1]}$$

- Как оптимизировать AUC напрямую?

⁶https://www.erikdrysdale.com/auc_max/

Оптимизация AUC

- Эквив. определение - доля верно упорядоченных пар:

$$AUC = \frac{\sum_{(i,j): y_i=-1, y_j=1} \mathbb{I}[f_w(x_j) > f_w(x_i)]}{\#[i : y_i = -1] \#[j : y_j = 1]}$$

- Как оптимизировать AUC напрямую? Сможем после сглаживания⁶:

$$AUC = \frac{\sum_{(i,j): y_i=-1, y_j=1} S(g(x_j) - g(x_i))}{\#[i : y_i = -1] \#[j : y_j = 1]} \quad \forall \text{дифф. } \uparrow S(\cdot)$$

- Чаще всего используется

$$S(u) = \ln \sigma(u) = \ln \frac{1}{1 + e^{-u}} = -\ln(1 + e^{-u})$$

⁶https://www.erikdrysdale.com/auc_max/

Содержание

- 1 Постановка задачи
- 2 Формирование вектора признаков
- 3 Поточечные (pointwise) методы
- 4 Попарные (pairwise) методы
- 5 Списочные (listwise) методы**
- 6 Меры качества ранжирования

Списочные (listwise) методы

- Списочные (listwise) методы пытаются оптимизировать весь список документов целиком.
 - самые точные, т.к. оптимизируют всю выдачу документов.
- Проблема: метрики качества зависят от порядка, который дискретен и недифференцируем.
- Подходы списочных методов:
 - взвешивать пары (i, j) согласно списочной мере качества
 - LambdaRank, LambdaMART
 - оптимизация сглаженной меры качества списка
 - SoftRank оптимизирует SoftNDCG вместо NDCG.
 - дискретный список \rightarrow непрерывное пространство распределений перестановок
 - ListNet
 - дискретный порядок \rightarrow скрытая случ. величина со своим распределением
 - LambdaLoss

LambdaRank

- Пусть Q - мера качества списка
(в оригинале NDCG).
- Оценим ΔQ_{ij} при $x_i \leftrightarrow x_j$.
- RankNet \rightarrow LambdaRank:

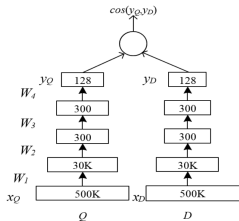
$$w := w + \frac{\varepsilon \alpha |\Delta Q_{ij}| (x_j - x_i)}{1 + e^{+\alpha (w^T x_j - w^T x_i)}}$$

- Изменение w выше для существенных изменений Q_{ij} .
 - на рисунке-черные: градиенты RankNet,
красные: градиенты LambdaRank (начало списка для Q важнее)
- Приближенно оптимизирует Q .



Модель DSSM⁷

- Модель DSSM (Deep Structured Semantic Model)
вычисляют соответствие
 - документа и запроса (ранжирование)
 - пользователя и товара (рек. система)
- Используется сиамская сеть, отображающее все в общее семантическое пространство.
 - cos-sim используется для оценки похожести



⁷ A Multi-View Deep Learning Approach for Cross Domain User Modeling in Recommendation Systems.

Модель DSSM

- Релевантность документа запросу:

$$R(Q, D) = \frac{\mathbf{y}_Q^T \mathbf{y}_D}{\|\mathbf{y}_Q\| \|\mathbf{y}_D\|}$$

- Вероятности клика на документы:

$$P(D_1, D_2, \dots | Q) = \text{softmax}_\tau (R(D_1, Q), R(D_2, Q), \dots)$$

- Оптимизация - максимизация вер-ти кликнутых док-тов⁸

$$\ln \prod_{(Q_i, D_i)} P(D_i | Q_i) \rightarrow \max_{\theta}$$

- Представление документов - через счётчик триграм:

- "курс \$" -> #ку, кур, урс, рс□, с□\$, □\$#

- работает с новыми словами!

⁸Это pointwise, pairwise или listwise подход?

Содержание

- 1 Постановка задачи
- 2 Формирование вектора признаков
- 3 Поточечные (pointwise) методы
- 4 Попарные (pairwise) методы
- 5 Списочные (listwise) методы
- 6 Меры качества ранжирования
 - Меры качества для бинарной релевантности
 - Меры качества для градаций релевантности

- 6 Меры качества ранжирования
 - Меры качества для бинарной релевантности
 - Меры качества для градаций релевантности

Меры качества для бинарной релевантности

- Рассмотрим бинарную разметку $y \in \{1, 0\}$ (релевантен/не релевантен) и запрос q среди всех запросов Q .
- На практике важны лишь первые N документов выдачи
- Предполагаем, y_1, y_2, \dots - упорядочены по $\downarrow f_w(x)$

Меры качества для бинарной релевантности

- Рассмотрим бинарную разметку $y \in \{1, 0\}$ (релевантен/не релевантен) и запрос q среди всех запросов Q .
- На практике важны лишь первые N документов выдачи
- Предполагаем, y_1, y_2, \dots - упорядочены по $\downarrow f_w(x)$
- Доля корректных пар = $AUC(q)$ [i, j - первые N док-тов]

$$ConcordantRatio(q) = \frac{|\{(i, j) : j \succ i \text{ и } f_w(x_j) > f_w(x_i)\}|}{|\{(i, j) : j \succ i\}|}$$

Меры качества для бинарной релевантности

- Рассмотрим бинарную разметку $y \in \{1, 0\}$ (релевантен/не релевантен) и запрос q среди всех запросов Q .
- На практике важны лишь первые N документов выдачи
- Предполагаем, y_1, y_2, \dots - упорядочены по $\downarrow f_w(x)$
- Доля корректных пар = $AUC(q)$ [i, j - первые N док-тов]

$$ConcordantRatio(q) = \frac{|\{(i, j) : j \succ i \text{ и } f_w(x_j) > f_w(x_i)\}|}{|\{(i, j) : j \succ i\}|}$$

$$Precision@N(q) = \frac{\sum_{n=1}^N y_n}{N} \text{ - не учитывает порядок}$$

Меры качества для бинарной релевантности

- Рассмотрим бинарную разметку $y \in \{1, 0\}$ (релевантен/не релевантен) и запрос q среди всех запросов Q .
- На практике важны лишь первые N документов выдачи
- Предполагаем, y_1, y_2, \dots - упорядочены по $\downarrow f_w(x)$
- Доля корректных пар = $AUC(q)$ [i, j - первые N док-тов]

$$ConcordantRatio(q) = \frac{|\{(i, j) : j \succ i \text{ и } f_w(x_j) > f_w(x_i)\}|}{|\{(i, j) : j \succ i\}|}$$

$$Precision@N(q) = \frac{\sum_{n=1}^N y_n}{N} \text{ - не учитывает порядок}$$

$$RR(q) = \frac{1}{\min_i y_i = 1}; \quad MRR = \frac{1}{|Q|} \sum_{q \in Q} RR(q) \text{ - mean reciprocal rank}$$

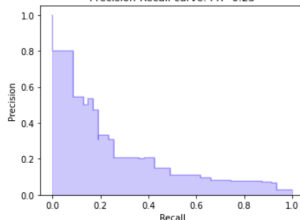
MRR на удивление хорошо ранжирует весь список на практике.

Меры качества для бинарной релевантности

- Средняя точность=площадь под Precision(Recall) кривой (зависимых от n среди первых N документов):

$$AP@N(q) = \frac{1}{\sum_{k=1}^N y_k} \sum_{n=1}^N y_n Prec@n(q)$$

Precision-Recall curve: PR=0.23



$$AP@N(q) = \sum_{n=1}^N (Recall@(n, N)(q) - Recall@(n-1, N)(q)) Prec@n(q)$$

Усредненная $AP@n(q)$ по запросам - mean average precision:

$$MAP@N = \frac{1}{|Q|} \sum_{q \in Q} AP@N(q)$$

Пример расчетов метрик качества

Результаты ранжирования:

n	$f_w(x)$	y	Prec@n
1	100	1	1
2	52	0	1/2
3	3	1	2/3
4	-200	1	3/4

- $RR = 1/1 = 1$
- $Recall@(1, 4) = 1/3$
- $Recall@(3, 4) = 2/3$.
- $AP@4 = \frac{1}{3} (1 + 2/3 + 3/4)$

- 6 Меры качества ранжирования
 - Меры качества для бинарной релевантности
 - Меры качества для градаций релевантности

Discounted Cumulative Gain

- Discounted Cumulative Gain

$$DCG@N(q) = \sum_{n=1}^N g(y_n) d(n)$$

- $g(y) \geq 0$ и \uparrow - поощрение за релевантность
- $d(n) \geq 0$ и \uparrow - дисконт за низкую позицию
- Классический выбор:

$$g(y) = 2^y - 1, \quad d(n) = \frac{1}{\log(i + 1)}$$

- $DCG@N(q) \geq 0$ неинтерпретируем и несравним для разных N , поэтому используют normalized DCG:

$$nDCG@N(q) = \frac{DCG@N(q)}{IdealDCG@N(q)} \in [0, 1]$$

Пример расчетов метрик качества

Результаты ранжирования:

n	$f_w(x)$	y	$g(y) = y^2$	$d(n) = 1/n$
1	100	3	9	1
2	52	4	16	1/2
3	3	0	0	1/3
4	-200	6	36	1/4

- $DCG@4(q) = 9 \cdot 1 + 16 \cdot \frac{1}{2} + 36 \cdot \frac{1}{4} = 9 + 8 + 9 = 26$
- $IdealDCG@4(q) = 36 \cdot 1 + 16 \cdot \frac{1}{2} + 9 \cdot \frac{1}{3} = 36 + 8 + 3 = 47$
- $nDCG@4(q) = 26/47$

pFound

- Мера качества pFound придумана в Яндексе.
- $y \in [0, 1]$ - вероятность найти ответ в документе
- p_{out} - вероятность прекратить поиск
- p_i - вероятность, что пользователь дойдет до i -й позиции выдачи.

pFound

- Мера качества pFound придумана в Яндексе.
- $y \in [0, 1]$ - вероятность найти ответ в документе
- p_{out} - вероятность прекратить поиск
- p_i - вероятность, что пользователь дойдет до i -й позиции выдачи. Считается рекуррентно:

$$p_1 = 1$$

$$p_{i+1} = p_i(1 - y_i)(1 - p_{out})$$

- Т.е. пользователь доходит до $(i + 1)$ -й позиции, если он дошел до i -й, не нашел ответ и не прекратил поиск.

$$pFound@N(q) = \sum_{n=1}^N p_n y_n$$

- В конце усредняем pFound по запросам.

Заключение

- Задача ранжирования - упорядочивание объектов по их релевантности поисковому запросу.
 - веб-страницы, ответы на вопросы, товары в магазине
- Признаки сочетают информацию о пользователе, запросе и документе.
 - PageRank-популярный способ рейтингования веб-страниц
- Методы ранжирования:
 - поточечные
 - попарные (самые популярные)
 - списочные (в теории - самые точные)
- Меры качества:
 - инвариантные к порядку среди первых N документов (precision, recall)
 - зависящие от порядка (nDCG)