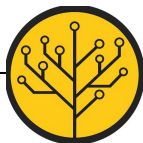


Ранжирование часть 2.



Факультет Компьютерных Наук, ПМИ
Кобелев Максим

Москва 2019



Оглавление

- Обзор основных подходов
- Поточечный подход на примере OC-SVM
- Попарный подход на примере RankSVM и RankNet
- Списочный подход на примере LambdaRank
- Заключение



Обзор основных подходов к ранжированию

В своей статье «Learning to Rank for Information Retrieval» и выступлениях на тематических конференциях, Тай-Ян Лью из Microsoft Research Asia проанализировал и выделил три подхода, в зависимости от используемого входного представления данных и функции штрафа:

Поточечный

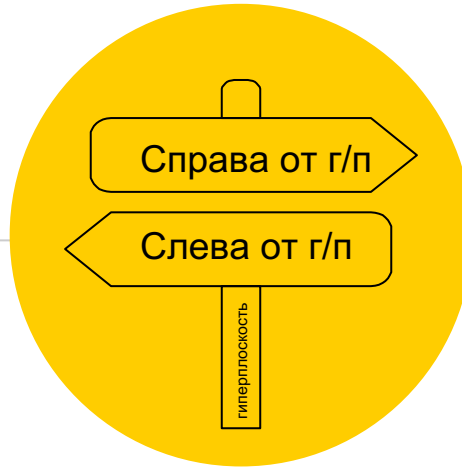
Пытаются приспособить методы классификации (ординальная) и регрессии тем самым восстанавливая численную оценку величины полезности этого документа.

Попарный

Строим функционал качества с учётом информации относительно пар объектов – документов.

Списочный

Сразу оценивается качество и происходит использование для оптимизации алгоритма информация всего списка документов



SVM

В качестве примера на каждом подходе
будем адаптировать алгоритм SVM под
задачу ранжирования.



Поточечный подход (Pointwise)

- Ordinal Classification SVM – адаптированная задача SVM классификации – базовый пример PointWise-подхода.
- Пусть у нас есть кое-какие прямые данные для обучения (т.е. про некоторые подмножества документов эксперт сказал, какие более релевантны, какие менее).
- Подход к решению: давайте обучать функцию, которая по данному вектору атрибутов $x \in \mathbb{R}^n$ выдаёт $f(x)$ и ранжирует документы по значению $f(x)$.

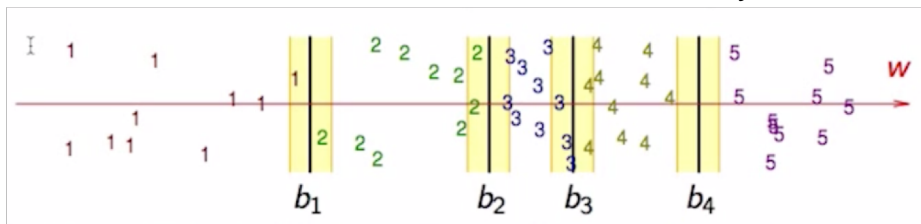


Поточечный подход (Pointwise)

Пусть $Y = \{1, \dots, K\}$, функция ранжирования *линейная* с порогоми

$$b_0 = -\inf, b_1, \dots, b_{K-1} \in \mathbb{R}, b_K = +\inf$$

$$a(x) = y, \quad \text{если} \quad b_{y-1} < \langle w, x \rangle \leq b_y$$



$$\left\{ \begin{array}{l} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l [y_i \neq 1] \cdot \xi_i + [y_i \neq K] \cdot \xi_i^* \rightarrow \min_{w, b, \xi} \\ b_{y_i-1} + 1 - \xi_i^* \leq \langle w, x_i \rangle \leq b_{y_i} - 1 + \xi_i \\ \xi_i^* \geq 0, \quad \xi_i \geq 0 \end{array} \right.$$



Попарный подход (Pairwise)

- RankSVM – адаптированная задача SVM классификации – базовый пример PairWise-подхода.
- Пусть у нас есть кое-какие прямые данные для обучения (т.е. про некоторые подмножества документов эксперт сказал, какие более релевантны, какие менее).
- Подход к решению: давайте обучать классификатор, который для каждой пары объектов умеет говорить какой из них лучше. Его цель – минимизировать количество неправильно отранжированных объектов.



Попарный подход (Pairwise)

- Переходим к гладкому функционалу ранжирования:

$$Q(a) = \sum_{i < j} [a(x_j) - a(x_i) < 0] \leq \mathcal{L}(a(x_j) - a(x_i)) \rightarrow \min,$$

$a(x)$ – алгоритм ранжирования

$\mathcal{L}(M)$ – убывающая непрерывная функция отступа $Margin(i, j)$

- Вспомнили про метрику “доли дефектных пар”.

$$\mathcal{L}(M) = (1 - M)_+ - \text{RankSVM}$$

$$\mathcal{L}(M) = \exp(-M) - \text{RankBoost}$$

$$\mathcal{L}(M) = \log(1 + e^{-M}) - \text{RankNet}$$



Попарный подход (Pairwise)

$$Q(a) = \frac{1}{2} \|w\|^2 + C \sum_{i < j} \mathcal{L}(a(x_j) - a(x_i)) \rightarrow \min_a$$

$a(x) = \langle w, x \rangle$ – функция ранжирования

$\mathcal{L}(M) = (1 - M)_+$ – функция потерь

$$M = \text{Margin}(i, j) = \langle w, x_j - x_i \rangle$$

Постановка оптимизируемого функционала:

$$\begin{cases} \frac{1}{2} \|w\|^2 + C \sum_{i < j} \xi_{ij} \rightarrow \min_{w, \xi} \\ \langle w, x_j - x_i \rangle \geq 1 - \xi_{ij}, & i < j \\ \xi_{ij} \geq 0, & i < j \end{cases}$$



От RankNet к LambdaRank

RankNet:

$$Q(a) = \sum_{i < j} \mathcal{L}(a(x_j) - a(x_i)) \rightarrow \min,$$

При $\mathcal{L}(M) = \log(1 + e^{-\delta M})$ и линейной модели $a(x) = \langle w, x \rangle$.

- Проблема с RankNet в том, что оптимизируется число попарных ошибок, а это не всегда то что нужно.
- Градиенты RankNet – это не то же самое, что градиенты nDCG.
- Так как же оптимизировать оптимизировать, скажем, nDCG?



Списочный подход (ListWise)

Stochastic Gradient Descend:

Выбираем на каждой итерации q , $i < j$ случайно.

$$w_q = w_{q-1} + \eta \cdot \frac{\delta}{1 + \exp(\delta \langle w, x_j - x_i \rangle)} \cdot (x_j - x_i)$$

Для оптимизации негладких функционалов MAP, nDCG, pFound достаточно домножить размер шага SGD на величину изменения данного функционала при перестановке элементов $x_i \leftrightarrow x_j$.

LambdaRank: домножение шага на изменение nDCG приводит к оптимизации nDCG.

$$w_q = w_{q-1} + \eta \cdot |\Delta nCDG_{ij}| \cdot \frac{\delta}{1 + \exp(\delta \langle w, x_j - x_i \rangle)} \cdot (x_j - x_i)$$

50.000

Оценок ассессоров добавляются в выборку ежегодно.

MatrixNet

Используется для построения формулы ранжирования, призвана увеличить количество критериев ранжирования, улучшить поиск. Настройка ранжирования сайта

PairWise

Этот подход выявлен на практике как лучший, даже несмотря на ListWise.





Источники

- Методы обучения ранжированию, [machinelearning](#)
- Задача ранжирования, [coursera](#)
- Learning to Rank, [wikipedia](#)
- From RankNet to LambdaRank to LambdaMART, [microsoft](#)



Основные идеи

Поточечный подход – Предполагается что каждой паре запрос–документ поставлена в соответствие численная оценка, либо её ординал.

Плохо – любой порядок объектов штрафуются функционалом качества одинаково. Штраф здесь зависит от величины ранга, а не от порядка.

Попарный подход – Пытаемся построить бинарный классификатор, который выясняет какой из двух поданных ему документов – лучше.

Плохо – функционал качества оценивает глобальный порядок, а не порядок для одной группы (list). Не учитываются зависимости в одной группе между сравниваемыми парами.

Списочный подход – Заключается в оценке качества и оптимизации сразу всего списка документов получая на выходе их перестановку.

Универсальность – Задача ранжирования позволяет приспособить (в зависимости от подхода) к себе почти любой алгоритм машинного обучения.