Анализ пользовательского поведения для уточнения релевантности документа в Web-поиске

Сафонова Ангелина Владимировна

научный руководитель
Гришкин В.М.

Введение в задачи ранжирования в веб-поиске

Задача поисковой машины не только находить необходимую информацию для пользователя но и предоставлять её в максимально удобном виде.

В веб поиске в ранжировании используются **факторы** основанные на

- Содержании документа(BM25)
- Структуре веб-графа(PageRank)
- Пользовательском поведении(CTR)

Ранжирование на основе пользовательских данных

Постановка задачи ранжирования на основе пользовательского поведения:

Максимально точно предсказать релевантность документа запросу только на основе данных поискового лога и ограниченного количества экспертных оценок

Этапы решения задачи ранжирования

- **Анализ исходных данных с целью выявления сигналов, полезных для ранжирования**
- **Построение пространства ранжирующих факторов**
- **Тренировка базовых моделей на основе поведенческих факторов с помощью SVM, подбор и верификация параметров**
- **Тренировка финальной ранжирующей модели на основе результатов базовых моделей с помощью SVMlight**
- **Оценка качества полученного ранжирования**

Описание исходных данных

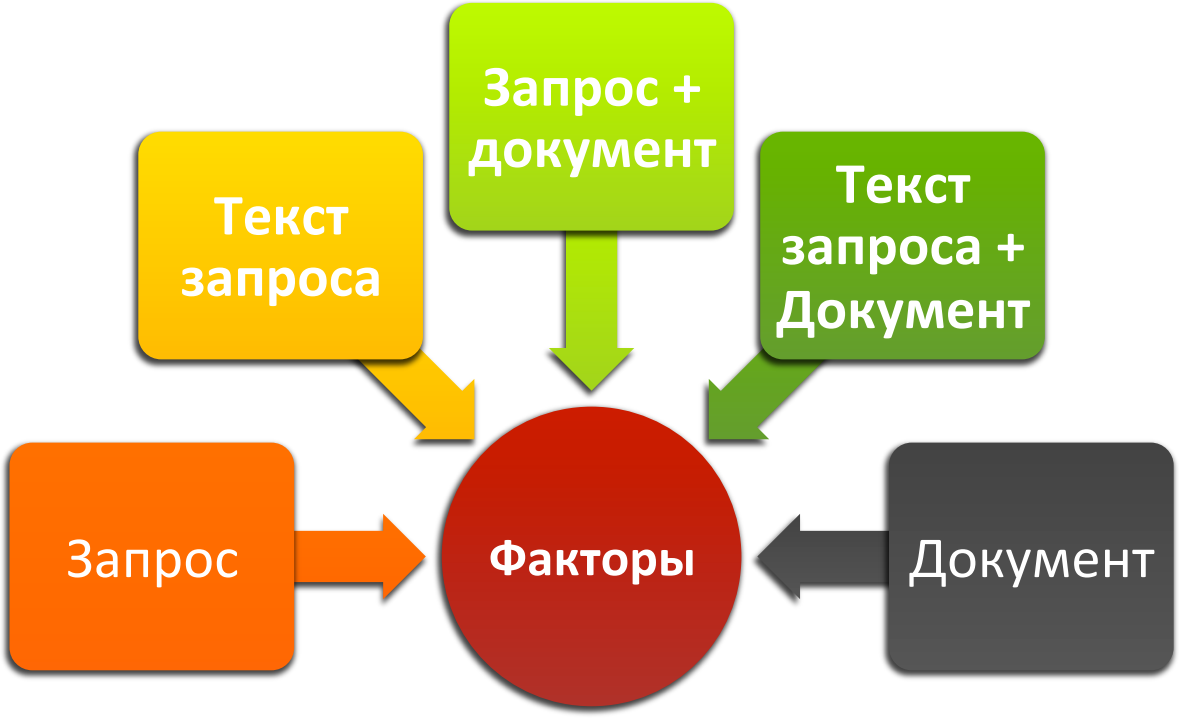
Исходные данные – анонимизированный поисковый лог, предоставлен компанией Яндекс в рамках конкурса Relevance Prediction Challenge(2011г.)

Поисковый лог – набор поисковых сессий, строк типа :

SessionID TimePassed TypeOfAction QueryID RegionID ListOfURLs
SessionID TimePassed TypeOfAction URLID
QueryID RegionID URLID RelevanceLabel (оценки релевантности)

Объём обрабатываемых данных – более 16Gb, содержит 30млн запросов, 117млн документов, 43млн поисковых сессий, 340млн действий, 71тыс бинарных оценок релевантности

Поведенческие факторы ранжирования

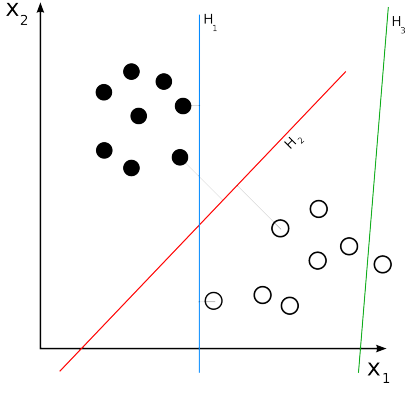


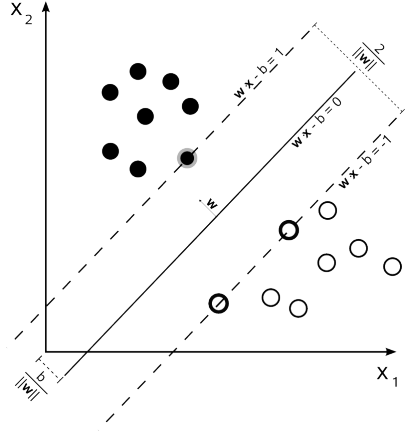
Получено 109 факторов ранжирования.

В топ10 вошли :

- Среднее время проведённое на документе по запросу
- Число показов/кликов документа по запросу
- Число стран в которых документ показывается/ кликается по запросу
- Число кликов по документу по всем запросам

SVM в задаче классификации





Рассмотрим задачу классификации множества на два непересекающихся класса, в которой объекты описываются *n*-мерными вещественными векторами: $X = \mathbb{R}^n$, $Y = \{1, -1\}$.

Необходимо построить линейный пороговый классификатор:

$$a(x) = sign\left(\sum_{j=1}^{n} w_j x^j - w_0\right) = sign(\langle w, x\rangle - w_0) \qquad (1)$$

где $x = (x^1, \dots, x^n)$ — признаковое описание объекта x;

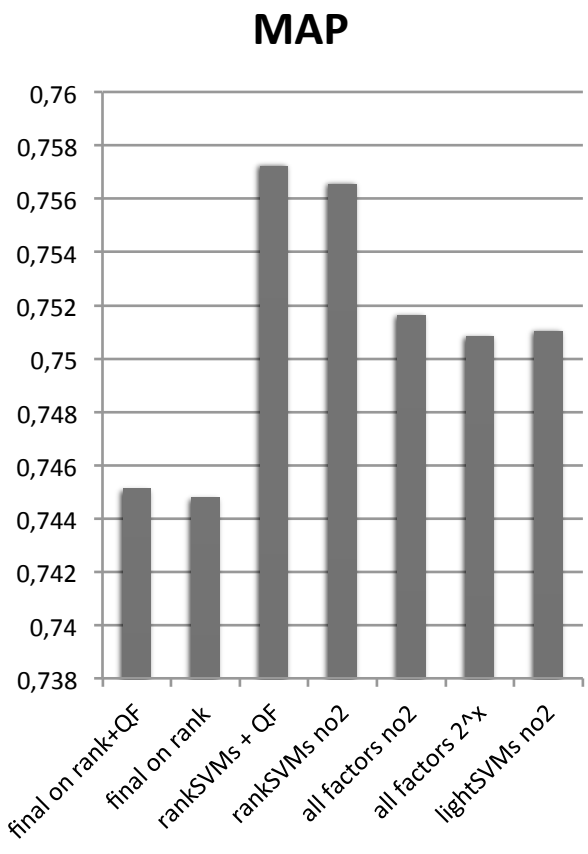
вектор $w = (w^1, \dots, w^n) \in \mathbb{R}^n$ и скалярный порог $w_0 \in \mathbb{R}^n$ являются параметрами алгоритма.

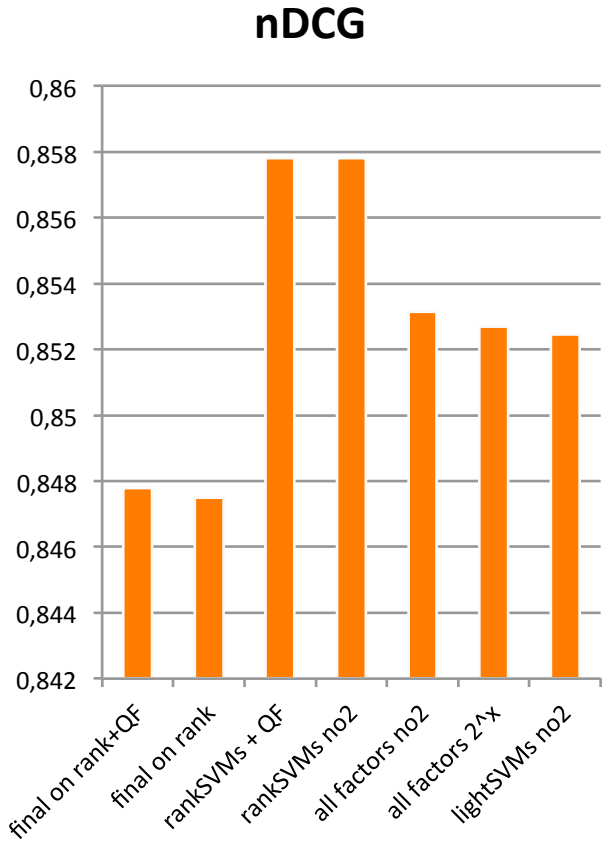
Уравнение $\langle w, x\rangle = w_0$ описывает гиперплоскость, разделяющую классы в пространстве $\mathbb{R}^n$.

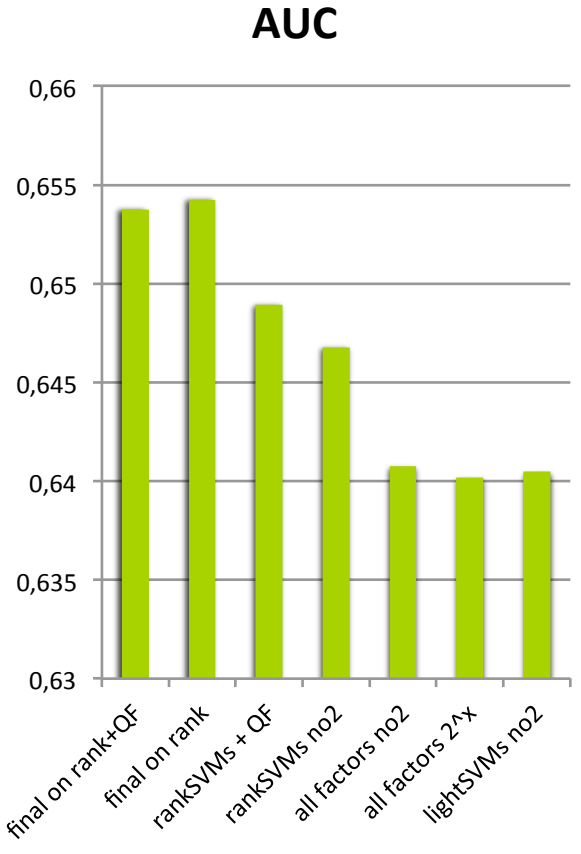
Реализация и использованные меры качества

- Система для расчёта факторов выполнена на языке программирования **Java**, среда разработки **IntellijIdea**
- В качестве рабочей реализации метода опорных векторов взята открытая библиотека **SVMlight V.6.02** (T.Joachims) написанная на C, и её специальная версия для обучения ранжированию **SVMrank V.1.00**.
- Аналитическая работа по исходным данным и оценка качества моделей выполнены на скриптовом языке **awk**
- Использованные меры качества: **MAP** (Mean Average Precision), **nDCG**(normalized Discounted Cummulative Gain), **AUC**(Area Under Curve)

Полученные результаты







Оценка качества полученных моделей

- В качестве лучшей итоговой модели была выбрана линейная модель, построенная SVMlight, на основе результатов 5 базовых ранжирующих функций, построенных SVMrank
- Минимально уступает ей по AUC(целевая метрика конкурса), однако превосходит по MAP и nDCG аналогичная модель с добавленными «запросными» факторами.
- В таблице приведено сравнение с возможным ранжированием по одному из пользовательских факторов,

MODEL	MAP	nDCG	AUC
best final validate	0,744804	0,847485	0,654213
41 num doc click	0,731767	0,831421	0,653706
76 querydoc show pos	0,730312	0,837621	0,625134
49 doc show pos	0,705871	0,818609	0,60262

Итоги работы

- **В обучении предложен оригинальный способ комбинации результатов сильных классификаторов, построенных с использованием SVM**
- **На основе пользовательских факторов получена ранжирующая модель, значительно превосходящая по используемым метрикам качества базовое ранжирование поисковой системы на тестовой выборке**
- **Произведён анализ пространства факторов, выделены перспективные направления развития**

Область применения и перспектива дальнейших исследований

Предложенная схема обучения применима:

- Внутри поисковой машины, в качестве метафактора или дополнительного ранжирования над основной моделью
- В метапоисковой системе для улучшения качества смешанной выдачи от нескольких поисковиков (требует оценок экспертов)
- В аналитической работе при разработке новых факторов, поиске новых сигналов, способствующих улучшению ранжирования

Перспектива дальнейших исследований - извлечение абсолютных оценок релевантности из пользовательского поведения и смешивания их с оценками полученными от экспертов.

Спасибо за Ваше внимание!

Описание мер качества

$$AUC = \frac{S_+ - n_+(n_+ + 1)/2}{n_+ n_-}$$

$$AveragePrecision = \frac{\sum_{k=1}^{n}(Precision(k) * Rel(k))}{N_{relevantDocuments}}$$

$$nDCG_p = \frac{DCG_p}{IDCG_p}, \qquad \text{где } DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{log_2 i}$$

SVM в случае линейной неразделимости

Чтобы обобщить метод опорных векторов на случай линейной неразделимости, позволим алгоритму допускать ошибки на обучающих объектах, но при этом постараемся сократить их количество. Введём набор дополнительных переменных $\xi_i \geq 0$, характеризующих величину ошибки на объектах x_i, $i = 1, \dots, l$. Возьмём за отправную точку задачу (3); смягчим в ней ограничения-неравенства, и одновременно введём в минимизируемый функционал штраф за суммарную ошибку:

$$\begin{cases} \dfrac{1}{2}\langle w, w\rangle + C\sum\limits_{i=1}^{l}\xi_i \to \min\limits_{w, w_0, \xi}; \\ y_i(\langle w, x_i\rangle - w_0) \geq 1 - \xi_i, \quad i = 1, \dots, l \\ \xi_i \geq 0, \quad i = 1, \dots, l; \end{cases} \tag{8}$$

Метод перекрёстной валидации

- Измерения качества полученных моделей проводилось по трём метрикам качества : AUC, nDCG и MAP. Представленные результаты усреднены по значениям, полученным на 5 независимых разбиениях обучающего множества методом кросс-валидации.

- Все экспериметны по подбору параметров, выбору типа обучения и настройке финальной модели ранжирования проводились исключительно в рамках обучающего множества данных(изначально предоставленных участникам конкурса). Только окончательные значения метрик качества вычислялись на тестовом множестве запросов.

Комбинация ранжирующих моделей

- *allfactors 2^x* – входят все 10 моделей, значение каждой преобразовано *2x*
- *allfactors no2* – входят все 10 моделей, без преобразований их результатов
- *lightSVMs no2* – входят только 5 моделей полученных *SVMlight*
- *rankSVMs no2* – входят только 5 моделей полученных *SVMrank*
- *rankSVMs +QF* – входят только 5 моделей полученных *SVMrank*, и в дополнение к ним, используются 40 «запросных» факторов, которые не были использованы при тренировке *SVMrank*-моделей, из-за специфики настройки алгоритма на парных сравнениях внутри запроса.