Министерство образования Российской Федерации
Московский государственный университет
им. М. В. Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра «Математические методы прогнозирования»

Дипломная работа:

«Прогнозирование вероятности кликов на новые баннеры»

Работу выполнил
Студент 517 группы
Колесников Александр Александрович

Научный руководитель:
д.ф.-м.н.
Воронцов Константин Вячеславович

Москва

2012

Содержание

1 Введение

Основным источником доходов поисковых систем является контекстная реклама. Главная задача системы показов контекстной рекламы — это отбор баннеров для размещения на странице, которую просматривает пользователь. Целью такого отбора является оптимизация прибыли поисковой системы и конверсии рекламодателей.

Существующие системы показов рекламы можно разбить на два класса, в зависимости от того, за что происходит списание денег с рекламодателей. В первом классе систем списание денег происходит за каждый показ рекламного объявления. Во втором классе рекламодатели платят только за клики по их рекламе, сделанные пользователями. Задача предсказания вероятности кликов на новые баннеры, рассмотренная в данной работе, решается в рамках второго класса систем. Наиболее известные поисковые системы, такие как Bing, Google, Yandex и Yahoo придерживаются второго подхода, т.е. списывают деньги за клики.

Рассмотрим подробнее то, как работает баннерная система: как появляются новые баннеры и что они из себя представляют, как происходит их отбор, сколько и за что платит рекламодатель.

Рекламодатель, приходя в систему контекстной рекламы, размещает в ней некоторое множество текстовых объявлений. Объявление состоит из текста, который логически можно разделить на две части: заголовок и тело. Объединение заголовка и тела баннера будем просто называть текстом баннера. Рекламодатель также указывает ссылку, по которой перейдет пользователь, если кликнет по баннеру. К каждому текстовому сообщению рекламодатель должен привязать некоторое множество фраз. Фразы нужны для организации отбора баннеров для дальнейшего показа. В случае, если фраза баннера полностью содержится в поисковом запросе пользователя, то баннер становится кандидатом на показ. И последнее, что необходимо указать рекламодателю — это ставки. Ставка указывается для каждой пары баннер-фраза и означает количество денег, которые готов заплатить рекламодатель за клик по баннеру, в случае, если он был отобран на показ по соответствующей фразе.

Рассмотрим теперь процесс отбора баннеров. Когда пользователь вводит запрос в поисковую систему, начинается поиск баннеров для показа на странице поисковой выдачи. Сначала, на первом этапе отбора, отбираются те баннеры, фразы которых содержатся в тексте запроса. Важно отметить, что каждому отобранному баннеру на этом этапе соответствует фраза, по которой он был отобран. В дальнейшем, чтобы не загромождать текст работы, каждую пару баннер-фраза будем называть просто баннером. Т.е. если к одному баннеру привязано пять различных фраз, то ему будет соответствовать пять различных баннеров. Часто оказывается, что баннеров-кандидатов на показ больше, чем рекламных мест на странице. В этом случае

необходимо отобрать наиболее подходящих кандидатов на показ. Достаточно естественным и широко используемым подходом в данном случае является отбор баннеров, имеющих наибольшее произведение ставки на вероятность клика по нему. Экономический смысл произведения вероятности клика и ставки — это ожидаемая прибыль от того, что мы покажем соответствующий баннер. Из сказанного выше становится понятно, что предсказание вероятности клика на баннер является важной задачей, решение которой необходимо для организация эффективного отбора баннеров. В англоязычной литературе для обозначения термина «вероятность клика на баннер» используется термин «Click-trough rate» или сокращенно — CTR.

В случае, если по баннеру собрана достаточно большая статистика и если сделать предположение, что клики и неклики по баннеру принадлежат биномиальному распределению, то можно воспользоваться оценкой максимального правдоподобия для оценки вероятности клика, которая равна отношению количества кликов по баннеру к количеству показов. К сожалению, для надежной оценки требуется собрать достаточно длинную историю показов баннера. Например, чтобы оценить CTR баннера, имеющего реальный CTR 10%, с точностью плюс/минус 1% и надежностью 90%, необходимо показать его 2600 раз. Это означает, что при цене за клик равной 0.50$, рекламодатель должен потратить около 1300$ перед тем, как CTR его баннера надежно стабилизируется около истинного значения.

В свете сказанного выше очень важно уметь оценивать CTR новых баннеров. Стартовый прогноз должен уменьшать экономические потери от неправильного предсказания CTR новых баннеров.

В работе [4] задача предсказания стартового CTR решается на основе кластеризации баннеров относительно их фраз. Авторы статьи выдвигают гипотезу, что схожие по фразам баннеры имеют схожий CTR. Далее все фразы разбивается на кластеры на основе текстовой близости с помощью алгоритма агломеративной кластеризации. В итоге мы получаем иерархию кластеров, где у каждой фразы есть непосредственный кластер, в котором она лежит, родитель этого кластера, прародитель и т.д. Для предсказания CTR баннера авторы статьи использовали такие признаки как средний CTR кластера, в котором он лежит, а также средний CTR кластеров, являющихся его родителями. Такой подход дал значительное улучшение по метрике среднеквадратичное отклонение предсказанного CTR от наблюдаемого. Основной вывод, сделанный в данной статье, заключается в том, что средний CTR кластеров схожих по фразе баннеров можно использовать для построения стартового прогноза.

Работа [5], появившаяся на год позже описанной выше статьи, предлагает гораздо более широкий подход к предсказанию CTR новых баннеров. В качестве выборки для обучения используется корпус баннеров, которые набрали не менее 100 показов в своей истории. Целевой

функцией при обучении является наблюдаемый CTR, т.е. отношения количества кликов по баннеру к количеству показов. В качестве регрессионной модели предсказания авторы используют логистическую регрессию:

$$\text{CTR} = \frac{1}{1 + \exp^{-Z}}; \quad Z = \sum_i w_i f_i(\text{ad}) \tag{1}$$

где $f_i(\text{ad})$ — это i-ый признак баннера ad, а w_i — коэффицент соответствующего признака в линейной модели. Используются следующие группы признаков : CTR по фразе, CTR по схожим фразам, признаки, описывающие внешний вид баннера, CTR домена, на который ссылается баннер, релевантность фразы тексту баннера и другие. Такой подход значимо улучшил качество первого описанного метода.

В данной работе, в отличие от предшествующих работ, предлагается несколько нововведений. Во-первых, в качестве целевой функции, вместо CTR баннеров с длинной историей, используются клики и неклики по баннеру из истории его показов (1, если клик был; 0 иначе). Т.е. объектами в выборке являются не баннеры, а отдельные показы баннеров. Такой подход имеет несколько преимуществ. Прежде всего, таким образом мы избавляемся от сдвига в выборке, вызванного тем, что мы обучение производится на баннерах с длинной историей: эмпирически известно, что баннеры, которые набирают большое количество показов, имеют больший средний CTR, чем в среднем все баннеры по системе. Также, в рамках такого подхода получается собирать бо̀льшие выборки. Во-вторых, в качестве модели предсказания я использовал взвешенную сумму деревьев регрессии. Такая модель заметно улучшила качество предсказания по сравнению с линейной моделью. В-третьих, было добавлено и эффективно реализовано вычисление ряда новых признаков, которые улучшили качество модели. Самые сильные новые признаки основаны на гибкой формализации понятия схожести фраз и текстов баннеров.

2 Данные для обучения

Для построения модели предсказания использовались данные сервиса контекстной рекламы Яндекс.Директ. Данные представляют из себя информацию о баннерах и соответствующих им показах и кликах за фиксированный промежуток времени. Важно отметить, что система Яндекс.Директ показывает контекстную рекламу вместе с результатами поиска и реклама располагается в двух различных блоках. Первый блок располагается непосредственно над результатами поиска и содержит не более трех объявлений(спецразмещение). Второй блок располагается справа от результатов поиска и содержит не более 8 объявлений(блок справа). Два этих блока имеют существенно разный средний CTR и для предсказания CTR в них используются

разные модели: в том числе отдельно копится статистика по показам и кликам. В данной работе рассмотрено построение стартового CTR для показов в блоке, расположенном над результатами поисковой выдачи, т.е. в спецразмещении.

Объектами в собираемой выборке являются показы. Каждому показу соответствует баннер и действие пользователя (1, если клик был; 0 иначе). Целевой функцией в данной выборке является действие пользователя. Заметим, что один и тот же баннер может быть показан много раз. Показы одного и того же баннера соответствуют идентичным описаниям в нашей выборке. Их предлагается объединять в один объект с весом, равным количеству таких событий.

Для удобства данные для обучения собираются в 2 этапа. На первом этапе данные из логов агрегируются в удобный и максимально краткий вид, необходимый для вычисления всех факторов и построения обучающей выборки. На втором этапе строится выборка, непосредственно используемая для обучения.

Опишем, какие данные агрегируются на первом этапе. События, произошедшие за фиксированный промежуток времени, агрегируются по соответствующим им **баннерам** и получается таблица, содержащая следующие данные:

- Количество показов **баннера** в спецразмещении за фиксированный промежуток времени, при которых был клик;
- Количество показов **баннера** в спецразмещении за фиксированный промежуток времени, при которых клика не было;
- Количество показов **баннера** в блоке справа за всю историю баннера;
- Количество кликов по **баннеру** в блоке справа за всю историю баннера;
- История домена, на который ссылается **баннер**: агрегированное количество кликов и показов в спецразмещении; всех баннеров, ссылающихся на соответствующих домен
- Заголовок **баннера**;
- Текст **баннера**;
- Текст фразы **баннера**.

Заметим, что в построенных данных нет исторической информации о кликах в спецразмещении; есть только клики за небольшой промежуток времени, на которые будет настраиваться регрессионная модель.

На основе описанных выше данных вычисляются признаки и строится обучающая выборка. Каждому объекту в описанных выше данных соотвествует 2 объекта в обучающей выборке. Первый объект имеет целевую функцию равную 1, вес равный количеству кликов по соответствующему баннеру и признаки, посчитанные для этого баннера. А второй объект имеет целе-

вую функцию равную 0, вес равный количеству некликов по баннеру и тот же самый набор признаков.

Поясним описанную структуру данных на примере. Пусть мы собираем выборку за фиксированную неделю w. Допустим, у нас в системе есть всего 2 баннера: B_1 и B_2. Они имеют следующие тексты, заголовки и фразы соответсвенно: (t_1, b_1, p_1) и (t_2, b_2, p_2). Про B_1 известно, что за неделю w по нему было совершено cl_1 кликов, и not_cliks_1 раз он был показан, при том, что клика не было. Для B_2 соответствующие числа равны cl_2 и not_clciks_2. Также, за все время жизни баннеров B_1 и B_2 известно их история показов справа: rsh_1, rcl_1 и rsh_2, rcl_2; а также история по их доменам: dsh_1, dcl_1 и dsh_2, dcl_2.

Тогда, собранные на первом этапе данные будут выглядить так:

Клики	Неклики	Показы справ	Клики справа	Показы дом.	Клики дом.	Загол.	Тело	Фраза
cl_1	not_cl_1	rsh_1	rcl_1	dsh_1	dcl_1	t_1	b_1	p_1
cl_2	not_cl_2	rsh_2	rcl_2	dsh_2	dcl_2	t_2	b_2	p_2

А обучающая выборка на основе этих данных примет следующий вид:

Клик	Вес	Фактор 1	Фактор 2	...	Фактор n
1	cl_1	f_1^1	f_2^1	...	f_n^1
0	not_cl_1	f_1^1	f_2^1	...	f_n^1
1	cl_2	f_1^2	f_2^2	...	f_n^2
0	not_cl_2	f_1^2	f_2^2	...	f_n^2

3 Постановка задачи и функционалы качества

Задачу, которая решается в данной работе, можно разбить на две подзадачи:

- Первая подзадача заключается в вычислении информативных признаков с точки зрения предсказания стартового прогноза CTR.
- Вторая подзадача заключается в построении регрессионной модели для предсказания CTR на основе вычисленных признаков.

Признаки и модель, использовавшиеся для построения регрессии, описаны в следующих главах.

Сейчас подробнее остановимся на функционалах качества. Пусть у нас есть обучающая выборка длины N в формате, описанном в предыдущем разделе. Пусть, также, $\{y_i\}_{i=1}^N$ — это последовательность кликов и не кликов из этой выборки, $\{w_i\}_{i=1}^N$ — соответствующие им веса, а $\{\text{ctr}\}_{i=1}^N$ — предсказанный CTR.

Для того, чтобы следить за качеством прогноза предлагается использовать два функционала: среднеквадратичное отклонение и линейную корреляцию Пирсона.

$$\begin{aligned} \text{MSE} &= \frac{\sum\limits_{i=1}^{N} w_i (y_i - \text{ctr}_i)^2}{N} \\ \text{LinearCorreleation} &= \frac{\sum_{i=1}^{N} (y_i - \bar{y})(\text{ctr}_i - \overline{\text{ctr}})}{\sqrt{\sum_{i=1}^{N} (y_i - \bar{y})^2 \sum_{i=1}^{N} (\text{ctr}_i - \overline{\text{ctr}})^2}} \end{aligned} \tag{2}$$

Эмпирически известно, что прирост линейной корреляции прогноза с кликами, хорошо согласуется с приростом экономических показателей баннерной системы. Стоит заметить, что линейная корреляция инвариантна относительно любого линейного преобразования прогноза CTR. Это опасно тем, что у нового прогноза может измениться среднее значение. Поэтому очень важно вместе с линейной корреляцией следить за MSE. Итого, критерий качества стартового прогноза CTR формулируется следующим образом: максимально улучшить линейную корреляции, не ухудшив среднеквадратичное отклонение.

4 Регрессионная модель

В качестве модели для предсказания использовалась взвешенная сумма деревьев регрессии, построенная методом градиентного бустинга [2]. Рассмотрим формальное описание метода градиентного бустинга. Пусть дана выборка $X = \{x_i, y_i\}_1^N$, где x_i принадлежит пространству объектов, а y_i — пространству ответов. Наша задача найти функцию $F(x)$, действующую из пространства объектов в пространство ответов, такую, чтобы минимизировать суммарную ошибку:

$$Loss = \sum_{i=1}^{N} \Psi(y_i, F(x_i)), \tag{3}$$

где $\Psi(y, F)$ - функция потерь.

Функцию $F(x)$ будем строить итеративно для $m = 1, 2, \ldots, M$. На каждой итерации m будем добавлять к уже построенной функции F дерево регрессии [1] $\beta_m h(x; a_m)$, где a_m — параметр, определяющий дерево регрессии, а именно предикаты в его вершинах и значения в листьях; β_m — вес дерева. При этом все добавляемые деревья имеют одинаковую форму, которая фиксируется заранее. Параметр a настраивается так, чтобы минимизировать функционал ошибки:

$$(\beta_m, a_m) = \underset{\beta, a}{argmin} \sum_{i=1}^{N} \Psi(y_i, F_{m-1}(x_i) + \beta h(x_i, a)) \tag{4}$$

$$F_m(x) = F_{m-1}(x) + \beta_m h(x, a_m) \tag{5}$$

Для оптимизации 4 воспользуемся методом градиентного спуска, т.е. параметр a будем искать так, чтобы $h(x; a)$ было как можно ближе по направлению к антиградиенту функционала

ошибки:

$$
\begin{aligned}
a_m &= \underset{a,\beta}{argmin} \sum_{i=1}^{N} [-g_m(x_i) - \beta h(x_i, a)]^2 \\
g_m(x_i) &= \left[\frac{\partial \Psi(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)}, i = 1 \dots N
\end{aligned}
\tag{6}
$$

Осталось вычислить оптимальную длину шага :

$$
\rho_m = \underset{\rho}{argmin} \sum_{i=1}^{N} \Psi(y_i, F_{m-1}(x_i) + \rho h(x_i, a_m)) \tag{7}
$$

Теперь можно обновить функцию $F(x)$:

$$
F_m(x) = F_{m-1}(x) + \rho_m h(x, a_m) \tag{8}
$$

Алгоритм 1 Общая схема градиентного бустинга

1: $F_0(x) = \underset{\rho}{argmin} \sum_{i=1}^{N} \Psi(y_i, \rho)$
2: **for** $m = 1, \dots, M$ **do**
3: $\quad g_{mi} = - \left[\frac{\partial \Psi(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)}, i = 1 \dots N$
4: $\quad (a_m, \beta) = \underset{a,\beta}{argmin} \sum_{i=1}^{N} [-g_{mi} - \beta h(x_i, a)]^2$
5: $\quad \rho_m = \underset{\rho}{argmin} \sum_{i=1}^{N} \Psi(y_i, F_{m-1}(x_i) + \rho h(x_i, a_m))$
6: $\quad F_m(x) = F_{m-1}(x) + \rho_m h(x, a_m)$
7: **end for**

Для построения модели, предсказывающей стартовый CTR, будем максимизировать логарифм правдоподобия прогноза CTR. Прогноз CTR и функцию F свяжем сигмойдным преобразованием: $CTR = \frac{\exp(F)}{1 + \exp(F)}$. Тогда функция потерь принимает следующий вид:

$$
\Psi_{ll}(y, F) = y \log \left(\frac{\exp(F)}{1 + \exp(F)} \right) + (1 - y) \log \left(\frac{1}{1 + \exp(F)} \right) \tag{9}
$$

При этом получаем, что

$$
g_m(x_i) = y_i - \frac{\exp(F)}{1 + \exp(F)} \tag{10}
$$

4.1 Регуляризация модели

В задачах прогнозирования достижение высокого качества предсказания на обучающей выборке может контр-продуктивно сказываться на обобщающей способности алгоритма. Этот эффект называется переобучением. Для борьбы с переобучением используются различные методы

Алгоритм 2 Бустинг для предсказания CTR

1: $F_0(x) = \underset{\rho}{argmin} \sum_{i=1}^{N} \Phi(y_i, \rho)$
2: **for** $m = 1, \ldots, M$ **do**
3: $\quad g_{mi} = (y_i - \frac{\exp(F(x_i))}{1 + \exp(F(x_i))}), i = 1 \ldots N$
4: $\quad (a_m, \beta) = \underset{a,\beta}{argmin} \sum_{i=1}^{N} [-g_{mi} - \beta h(x_i, a)]^2$
5: $\quad \rho_m = \underset{\rho}{argmin} \sum_{i=1}^{N} \Psi_{ll}(y_i, F_{m-1}(x_i) + \rho h(x_i, a_m))$
6: $\quad F_m(x) = F_{m-1}(x) + \rho_m h(x, a_m)$
7: **end for**

регуляризации, которые добавляют ограничения в процесс обучения. Для аддитивной модели, которая описана выше, ограничение числа деревьев M является естественным и простым способом регуляризации. Оптимальное значение M можно оценить с помощью кросс-валидации.

Однако, на практике заметно лучшие результаты удается получить, если ввести еще один параметр регуляризации $\nu \in (0, 1]$, который немного модифицирует шаг, на котором к уже построенной функции F добавляется очередное дерево:

$$F_m(x) = F_{m-1}(x) + \nu \rho_m h(x, a_m). \tag{11}$$

Введение параметра ν в модель приводит к тому, что в модели оказывается два параметра регуляризации : M и ν. Каждый из них может контролировать переобученность и влияет на оптимальное значение другого. Чем меньше значение ν, тем, как правило, больше оптимальное значение M. В идеальном случае нужно провести оптимизацию по двум параметрам и выбрать наилучшую пару, но такая оптимизация сопряжена со значительными вычислительными затратами. Поэтому часто используется такая схема:

- Фиксируется множество значений ν. Для большинства задач достаточно зафиксировать множество $A_\nu = \{1.0, 0.5, 0.4, 0.3, 0.2, 0.1, 0.075, 0.05, 0.03, 0.02, 0.01, 0.005\}$.
- Для каждого $\nu \in A_\nu$ находим оптимально M и выбираем оптимальную пару.

5 Поиск схожих по фразе или тексту баннеров

В данном параграфе, который разбит на 3 части, рассказано о использующемся для построения факторов алгоритме поиска близких по тексту баннеров. В первой части описана векторная модель текстового поиска, которая является основой алгоритма. Во второй части

затронута проблема эффективной реализации векторной модели поиска. В третьей вводится точное понятие близких по тексту баннеров.

5.1 Векторная модель текстового поиска

В этом разделе приведен обзор векторной модели текстового поиска, т.к. именно она используется для установления меры сходства между текстами баннеров.

Пусть у нас есть множество текстов D. Каждый текст $d \in D$ представляет из себя некоторое множество термов $term_d$. Для каждой пары терм-текст введем величину $tf_{term,d}$ равную квадратному корню из количества раз, которое терм $term$ встретился в тексте d.

Далее введем характеристику терма, оценивающую насколько часто он встречается во множестве текстов D:

$$idf_{term} = 1 + \log \frac{|D|}{|\{d \in D | term \in d\}| + 1} \tag{12}$$

Пусть во множестве документов D содержится n различных термов. Каждому терму присвоим уникальный порядковый номер i(и обозначим его $term_i$), не превосходящий n Теперь каждому документу d можно сопоставить n-мерный вектор v_d такой, что :

- $v_d[j] = tf_{term_j,d} \cdot idf(term_j)$, если $term_j \in d$
- $v_d[j] = 0$, если $term_j \notin d$

После того, как мы представили документы в виде векторов, можно ввести функцию близости CosineSim документов $d1$ и $d2$ как косинус угла между соответствующими векторами :

$$\text{CosineSim}(d1, d2) = \frac{v_{d_1} \cdot v_{d_2}}{|v_{d_1}| \cdot |v_{d_2}|} \tag{13}$$

Для документов, не имеющих общих слов, эта функция примет значение равное 0. Для документов с идентичными текстами функция принимает значение равное 1. Во всех остальных случаях значение функции лежит в отрезке $(0.0, 1.0)$, при этом мы считаем, что, чем больше значение введенной функции близости, тем документы более похожи.

Более подробно о векторной модели поиска можно почитать в [3].

5.2 Эффективный поиск схожих баннеров

Поиск схожих по тексту достаточно трудоемкая операция. Действительно, пусть нам надо для каждого баннера из некоторой коллекции найти все схожие баннеры(согласно описанной

выше модели поиска) из этой же коллекции. Вычислительная сложность такой задачи при реализации, предусматривающей просмотр всех баннеров при поиске похожих будет расти квадратично по числу баннеров. Квадратичная вычислительная сложность становится неприемлимой, если количество обрабатываемых баннеров имеет порядок миллиона или более. Поэтому при больших выборках необходимо использовать эффективные алгоритмы, реализующие векторную модель текстового поиска.

Для реализации эффективного поиска схожих баннеров предлагается использовать инвертированный текстовый индекс. Подробное описание этой структуры данных и алгоритмов над ней можно найти в [3]. В качестве реализации этого подхода использовался поисковый сервер Solr[1] [6], предназначенный для полнотекстового поиска. Поиск в Solr основан на описанной выше векторной модели текстового поиска, В функциональсть Solr входит функция поиска заданного числа наиболее близких текстов в смысле описанной выше функции CosineSim.

5.3 Понятие схожих по фразе или тексту баннеров

Перед тем, как окончательно сформулировать понятие схожести баннеров, необходимо ввести несколько важных определений.

Вероятность клика на баннер очень сильно зависит от того, релевантен ли текст соответствующей ему фразы тексту самого баннера. Это обстоятельство можно учитывать при построении факторов, основанных на схожести текстов. В связи с этим для каждого баннера **ad** введем величину $\mathrm{rel}_{\mathrm{ad}}$.

Второе важное понятие — это кворум. Под кворумом мы будем понимать некоторую функцию $f : \mathbb{N} \to \mathbb{N}$, При этом f имеет следующий смысл: пусть текст T имеет n слов, тогда для любого схожего текста будем требовать, чтобы он содержал не менее $f(n)$ слов, присутствующих в тексте T.

Теперь все готово для того, чтобы ввести понятие схожести на множестве баннеров. Баннеры могут быть схожими относительно своих фраз или текстов. Введем схожесть относительно фраз. Схожесть относительно текстов вводится аналогично.

Итак, баннер ad_1 схож с баннером ad_2, если фраза phrase_1 баннера ad_1 схожа с фразой phrase_2 баннера ad_2. Фраза phrase_1 схожа с phrase_2, если

- В phrase_2 содержится кворум $f(\#\{\mathrm{phrase}_1\})$ слов из phrase_1, где $\#\{\mathrm{phrase}_1\}$ — кол-во слов в phrase_1
- $|\mathrm{rel}_{\mathrm{ad}_1} - \mathrm{rel}_{\mathrm{ad}_2}| < \Delta p$

[1] `http://lucene.apache.org/solr/`

- Если более N фраз из нашего корпуса удовлетворяет первым двум условиям, то отбирается топ N наиболее релевантных фраз (в смысле метрики CosineSim)

Введенное определение позволяет эффективно искать схожие баннеры на основе полнотекстовых индексов. Также, введенное определение является достаточно гибким. Оно зависит от четырех параметров: текста, по которому устанавливается схожесть, функции кворума, Δp, N. В качестве текста, по которому устанавливается схожесть, рекомендуется попробовать заголовок баннера, тело баннера, объединение заголовка и тела, фразу баннера. В данной работе для построения признаков, основанных на схожести, использовалось объединение заголовка и тела и фраза баннера. Добавление признаков, основанных на схожести заголовка и тела отдельно, не дало значимого прироста качества.

Множество баннеров, схожих с баннером ad относительно текста text, кворума $q : \mathbb{N} \to \mathbb{N}$, $\Delta p \in [0, 1]$ и $N \in \mathbb{N}$ будем обозначать как $\mathrm{Sim}^{\mathrm{ad}}(text, q, \Delta p, N)$, а множество CTR'ов похожих баннеров будем обозначать $\mathrm{CTRSim}^{\mathrm{ad}}(text, q, \Delta p, N)$.

6 Признаки

Все признаки, использовавшиеся для обучения, можно разбить на 6 групп. Рассмотрим отдельно каждую из них.

6.1 Признаки, описывающие фразу

Фраза, привязанная к баннеру, является одним из самых сильных факторов, влияющих на CTR баннера. Как правило, она определяет тематику запросов, по которым показывается баннер, намерения пользователя и т.п.

Для построения признаков, основанных на фразе, рассмотрим фразу как множество слов $\{\mathrm{word}_1, \mathrm{word}_2, \ldots, \mathrm{word}_k\}$. Количество k слов — сильный признак, влияющий на CTR баннера. В таблице 1 приведена зависимость среднего значения CTR от количества слов во фразе. В этой и во всех остальных таблицах реальные значения CTR заменены относительными из соображений секретности. Рост CTR вместе с ростом количества слов во фразе объясняется тем, запрос пользователя обязательно содержит все слова фразы баннера. Как следствие, длинные фразы «угадывают» больше слов запроса и в среднем точнее соответствуют тому, что ищет пользователь.

Заметим, что важно учитывать не только количество слов, но и сами слова. Например, слова «продажа» или «цена», которые часто встречаются во множестве всех фраз, несут в себе

Таблица 1. Зависимость CTR от кол-ва слов во фразе

Количество слов	Средний CTR
1	1.0000
2	1.1077
3 и более	1.3039

меньше информации, чем, например, слово «черепица» с точки зрения угадывания намерений пользователя. Предлагается каждому слову присвоить вес — его IDF. Чем больше IDF слова, тем реже оно встречается во множестве фраз и тем более информативно его присутствие во фразе. На основе весов слов фразы в модель были добавлены следующие факторы:

- Сумма IDF всех слов фразы;
- Произведение IDF всех слов фразы;
- Средний IDF всех слов фразы.

6.2 Признаки, описывающие релевантность фразы тексту баннера

Следующая важная группа факторов — это факторы, описывающие релевантность фразы тексту баннера. Понятно, что чем лучше фраза соответствует тексту баннера, тем выше должен быть CTR баннера. В таблице 2 отражена зависимость среднего CTR баннеров от текстовой релевантности фразы тексту баннера.

Таблица 2. Зависимость среднего CTR от релевантности фразы тексту баннера

Интервал значений признака	Средний CTR
$[0.0, 0.30]$	1.0000
$[0.30, 0.50]$	1.2637
$[0.50, 0.70]$	1.4583
$[0.70, 1.00]$	1.5992

Помимо текстовой релевантности предлагается использовать еще несколько признаков, для учета степени соответствия фразы тексту баннера:

- Индикатор того, что все слова фразы содержатся в тексте баннера
- Индикатор того, что хотя бы одно слово фразы содержится в тексте баннера

- Доля и количество слов фразы, которые содержатся в заголовке баннера
- Доля и количество слов фразы, которые содержатся в теле баннера
- Доля и количество слов фразы, которые содержатся в тексте баннера

6.3 Факторы, основанные на CTR схожих по фразе баннеров

В работе [4] было показано, что баннеры с одинаковыми или близкими фразами имеют схожий CTR. Основываясь на этом наблюдении предлагается ввести следующий признак f_{phrase} для баннера ad:

$$f = \frac{\sum_{\text{ctr} \in \text{CTRSim}^{\text{ad}}(\text{phrase}, q, \Delta p, N)} ctr}{\#\{\text{Sim}^{\text{ad}}(\text{phrase}, q, \Delta p, N)\}} \tag{14}$$

f_{phrase} имеет простой смысл — это средний CTR схожих баннеров. Этот признак зависит от трех параметров: кворума q, Δp и N. Рекомендуется подобрать адекватные значения отдельно для каждого параметра, в зависимости от специфики решаемой задачи. Затем перебрать всевозможные тройки этих параметров и выбрать такую тройку, которая максимизирует линейную корреляцию соответствующего ей признака и целевой функции кликов.

Таблица 3. Зависимость CTR от признака «средний CTR по фразе»

Интервал значений признака	Средний CTR
$[0.0, 0.03]$	1.0000
$[0.03, 0.05]$	1.1830
$[0.05, 0.08]$	1.4413
$[0.05, 0.08]$	1.4413
$[0.08, 0.10]$	1.9779
$[0.10, 0.12]$	2.2021
$[0.12, 0.15]$	2.2909
$[0.15, 0.17]$	2.6936
$[0.17, 0.20]$	3.0145
$[0.20, 1.00]$	3.3654

6.4 Факторы, основанные на CTR схожих по тексту баннеров

Подобно тому, как мы ввели признак, основанный на среднем CTR баннеров со схожими фразами, можно ввести признак, основанный на CTR баннеров со схожими текстами.

$$f_{\text{text}} = \frac{\sum_{\text{ctr} \in \text{CTRSim}^{\text{ad}}(\text{phrase}, q, \Delta p, N)} ctr}{\#\{\text{Sim}^{\text{ad}}(\text{phrase}, q, \Delta p, N)\}} \tag{15}$$

Таблица 4. Зависимость CTR от признака «средний CTR по тексту»

Интервал значений признака	Средний CTR
$[0.0, 0.03]$	1.0000
$[0.03, 0.05]$	1.3925
$[0.05, 0.08]$	1.9171
$[0.05, 0.08]$	1.9171
$[0.08, 0.10]$	2.3711
$[0.10, 0.12]$	3.0306
$[0.12, 0.15]$	3.3968
$[0.15, 0.17]$	3.8422
$[0.17, 0.20]$	4.4022
$[0.20, 1.00]$	5.1869

6.5 Статистика в блоке справа

Как уже говорилось выше, помимо блока «спецразмещение», для которого мы строим модель предсказания стартового CTR, есть еще «блок справа». Нередко встречаются баннеры, которые накопили длинную историю показов в «блоке справа», но почти не имеют истории в «спецразмещении». Поэтому добавим следующие признаки: количество кликов и показов в блоке «справа», CTR «справа». Ниже приведена таблица 5, которая показывает, что CTR в «блоке справа» хорошо коррелирует с CTR в «спецразмещении».

6.6 Статистика по домену

Баннеры, ссылающиеся на один и тот же домен, очень часто имеют схожие CTR. Основываясь на этом наблюдении, в модель были добавлены признаки, описывающие агрегированную статистику кликов и показов по всем баннерам, ссылающимся на домен баннера, CTR которого мы предсказываем.

Таблица 5. Зависимость CTR от CTR «справа»

Интервал значений признака	Средний CTR
$[0.0, 0.01]$	1.0000
$[0.01, 0.02]$	1.1597
$[0.02, 0.03]$	1.4280
$[0.03, 0.04]$	1.6059
$[0.04, 0.05]$	1.7700
$[0.05, 0.06]$	1.9340
$[0.06, 0.07]$	2.1250
$[0.07, 0.08]$	2.3550
$[0.08, 0.09]$	2.3950
$[0.09, 1.00]$	2.8325

7 Численные эксперименты

В этом разделе рассмотрены проведенные численные эксперименты.

Опишем данные, которые использовались для эксперимента. Сначала была собрана база рекламных объявлений с историей, превышающей 250 показов. Объем этой базы составил два миллиона баннеров. Эта база рекламных объявлений использовалась для построения признаков, основанных на CTR схожих баннеров.

Затем была собрана статистика показов за неделю и из нее была взята подвыборка, соответствующая показам одного миллиона случайно выбранных баннеров. Статистика показов и кликов по этим баннерам использовалась для построения обучающей выборки. Формат обучающей выборки описан выше в разделе 2.

На основе показов, которые произошли в системе на месяц позже показов, по которым была получена обучающая выборка, была собрана контрольная выборка. Контрольная выборка имеют такой же формат, как обучающая выборка.

Численные эксперименты на основе собранных данных проводились по одной схеме:

- По обучающей выборке строилась формула для предсказания стартового CTR. В различных экспериментах из выборки либо выкидывались некоторые признаки, либо фильтровалась часть событий;
- Затем по полученной формуле делался прогноз для баннеров из контроля;
- Полученный прогноз встраивался в текущую формулу прогнозирования CTR. Текущая

Таблица 6. Зависимость CTR от доменного CTR

Интервал значений признака	Средний CTR
$[0.0, 0.03]$	1.0000
$[0.03, 0.05]$	1.8407
$[0.05, 0.08]$	3.0796
$[0.05, 0.08]$	3.0796
$[0.08, 0.10]$	4.0973
$[0.10, 0.12]$	5.0664
$[0.12, 0.15]$	6.3850
$[0.15, 0.17]$	7.4513
$[0.17, 0.20]$	8.8451
$[0.20, 1.00]$	11.2389

формула основывается на множестве признаков, самым сильным из которых является сглаженное со средним CTR по системе отношение количества кликов к показам в истории баннера. Встраивание прогноза в формулу заключалось в том, что сглаживание при вычислении этого признака стало проводиться со стартовым прогнозом, а не константой, равной среднему CTR по системе;

- На контрольной выборке замерялся прирост линейной корреляции новой формулы предсказания CTR по сравнению с текущей. При этом прирост измерялся как в целом по всем баннерам, так и по различным срезам, которые соответствовали баннерам с разной длиной истории показов. Использовались следующие срезы:
  - баннеры, у которых в истории от 0 до 10 показов,
  - баннеры, у которых в истории от 10 до 20 показов,
  - баннеры, у которых в истории от 20 до 50 показов,
  - баннеры, у которых в истории от 50 до 100 показов,
  - баннеры, у которых в истории от 100 до 200 показов,
  - все баннеры;

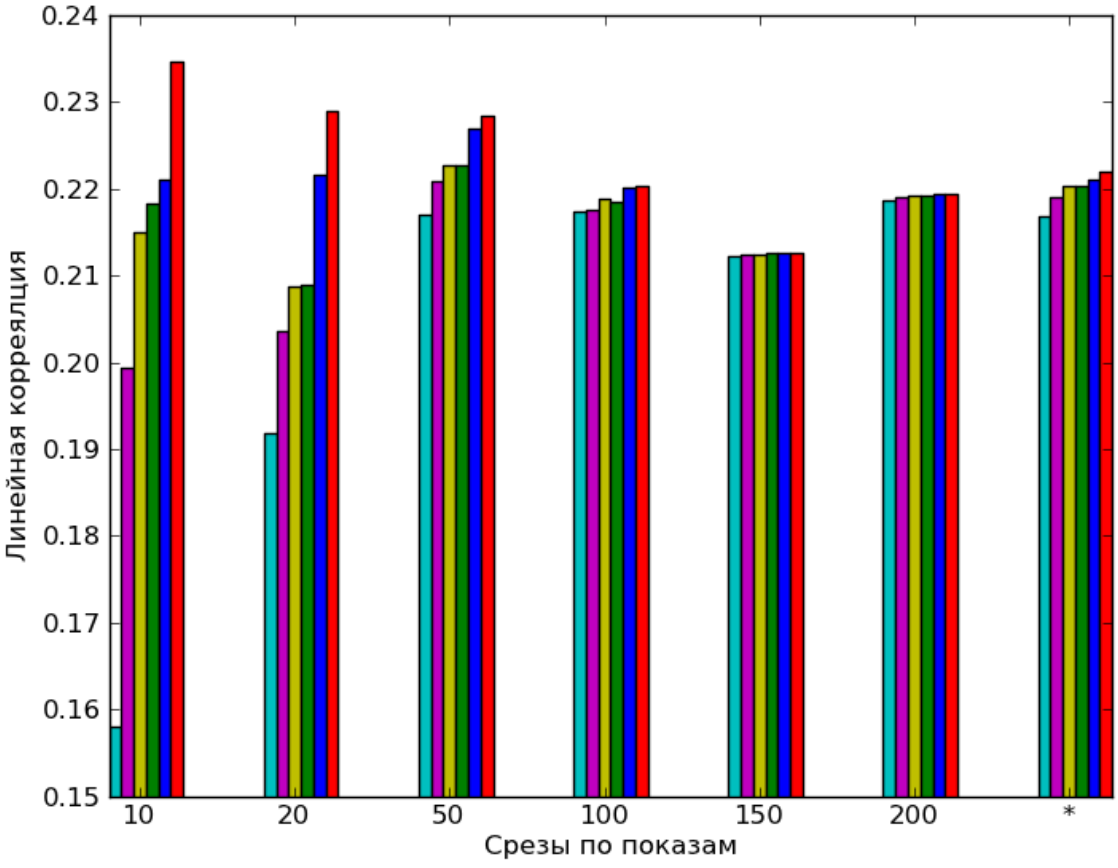
  Понятно, что значимого прироста качества следует ожидать на новых только на срезах, которые соответствуют новым баннерам.

7.1 Группы признаков

Рассмотрим влияние рассмотренных выше групп признаков на качество формулы. Наборы признаков будем добавлять к модели в том порядке, в котором они описаны выше. Результаты эксперимента представлены на диаграмме 1. По горизонтальной оси отложены различные срезы по показам (в том порядке, в котором они описаны выше). В каждом срезе 6 столбиков, которые соответствуют формуле предсказания, которая обучалась на первой группе признаков, на первых двух, на первых трех и т.д.

На рассматриваемой диаграмме особенно показателен первый срез, который соответствует самым новым баннерам, набравшим не более 10 показов в истории. Видно, каждая новая группа улучшает качество прогноза. Если за базовую уровень качества прогноза взять результат формулы, обученной на первой группе признаков, то получается, что добавление второй группы улучшает качество на 26.1%, третьей — на 36.0%, четвертой — на 38.2%, пятой — на 39.9%, а шестой — на 48.5%.

Рис. 1



7.2 Отбор признаков

Для того, чтобы описать процесс отбора признаков, необходимо сначала ввести понятие «влияние признака», которое будем обозначать как effect.

Итак, пусть мы построили регрессионную формулу, состоящую из суммы M деревьев регрессии по обучающей выборке X. При этом i-ое дерево обозначим как h_i. Каждое h_i дерево содержит p предикатов. Деревья регрессии будем рассматривать как функции от p предикатов: $h_i(g_{i1}(x), g_{i2}(x), \ldots, g_{ip}(x))$.

Для каждого $i \in [1, M]$ и двоичного вектора $x = (x_1, x_2, \ldots, x_p)$ введем функцию $n_i(x)$ — количество объектов $x \in X$ таких, что $x_1 = g_{i1}(x), x_2 = g_{i2}(x), \ldots, x_p = g_{ip}(x)$

Теперь опишем схему подсчета ненормированного влияния Effect_k от k-ого признака:

- $\text{Effect}_k := 0$
- Для каждого вхождения фактора k в дерево i в качестве признака, используемого в предикате под номером a делаем следующий шаг:
- Для всех $x_1 \in \{0,1\}, x_2 \in \{0,1\}, \ldots, x_{a-1} \in \{0,1\}, x_{a+1} \in \{0,1\}, \ldots, x_p \in \{0,1\}$

$$\begin{aligned} v_1 &= h_i(x_1, x_2, \ldots, x_{a-1}, 0, a_{a+1}, \ldots, x_p) \\ v_2 &= h_i(x_1, x_2, \ldots, x_{a-1}, 1, a_{a+1}, \ldots, x_p) \\ c_1 &= n_i(x_1, x_2, \ldots, x_{a-1}, 0, a_{a+1}, \ldots, x_p) \\ c_2 &= n_i(x_1, x_2, \ldots, x_{a-1}, 1, a_{a+1}, \ldots, x_p) \end{aligned} \tag{16}$$

Если $c_1 > 0$ или $c_2 > 0$, то

$$\text{Effect}_k := \text{Effect}_k + (v_1 - \frac{v_1 c_1 + v_2 c_2}{c_1 + c_2})^2 c_1 + (v_2 - \frac{v_1 c_1 + v_2 c_2}{c_1 + c_2})^2 c_2 \tag{17}$$

Нормированное влияние вычисляется по формуле $\text{effect}_k = \dfrac{\text{Effect}_k}{E}$, где $E = \sum_k Effect_k$.

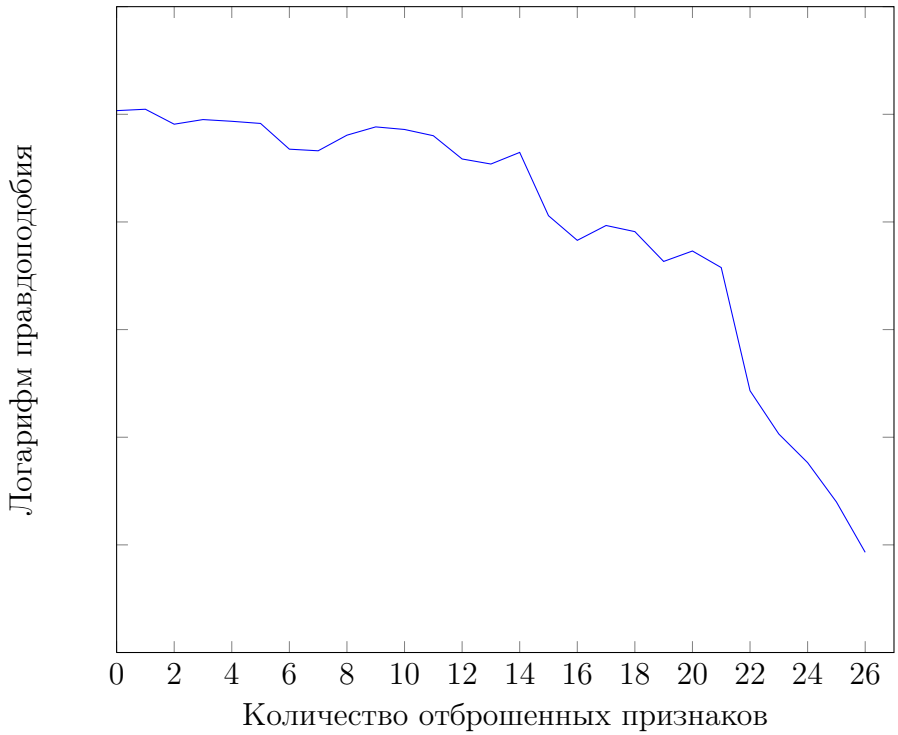
Отбор признаков был проведен следующим образом:

- Обучаем регрессионную формулу на всех признаках;
- После обучения отбрасываем признак с наименьшим влиянием;
- Обучаемся заново на новом множестве признаков;
- Повторяем предыдущие 2 шага, пока не отбросим все признаки.

На рисунке 2 приведена зависимость значения функционала качества модели от количества отброшенных признаков. Около 14 признаков получается отбросить без значимой потери

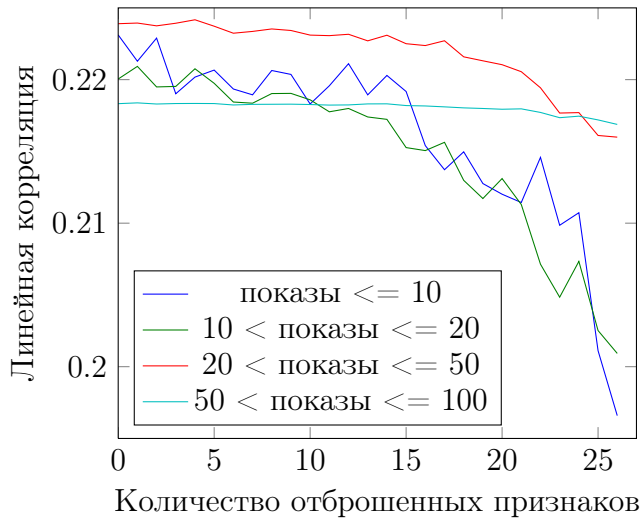
качества. Далее качество модели начинает заметно ухудшаться. Стоит отметить что среди 14 признаков, которые удается отбросить без потери качества, нет такого множества признаков, которые образовали одну из 6 описанных выше групп.

Рис. 2. Зависимость функционала качества от количества отброшенных признаков



На рисунке 3 приведена зависимость линейной корреляции прогноза и кликов на контрольных данных в различных срезах, в зависимости от числа показов. Поведение линейной корреляции на срезах, соответствующих новым баннерам, сходно с поведением функционала качества на обучении. На баннеры с историей стартовый прогноз практически не влияет.

Рис. 3. Зависимость линейной корреляции от количества отброшенных признаков



7.3 Подвыборки для обучения на основе истории показов

Если посчитать средний CTR баннеров с короткой историей и баннеров с длинной историей, то окажется, что баннеры с длинной историей имеют больший средний CTR. Это связано с тем, что баннеры, которые показываются по частотным запросам и за которые рекламодатели готовы платить большие суммы денег, как правило оказываются достаточно высокого качества, имеют высокую релевантность и, как следствие, высокий CTR. Такие баннеры могут значительно искажать выборку, т.к. набирают имеют высокую долю показов среди всех показов и не являются типичными представителями баннеров, которые добавляются в систему.

В связи с этим замечанием предлагается фильтровать баннеры в выборке на основе длины их истории. Были попробованы фильтры, которые отбрасывают баннеры, имеющие более 10, 20, 30, 50, 75, 100, 150, 200 показов. Такая фильтрация помогла улучшить качество прогноза на контроле на 5% среди баннеров с историей от 0 до 10 показов. Лучшие результаты показал фильтр, который отбрасывает баннеры имеющие более 100 показов.

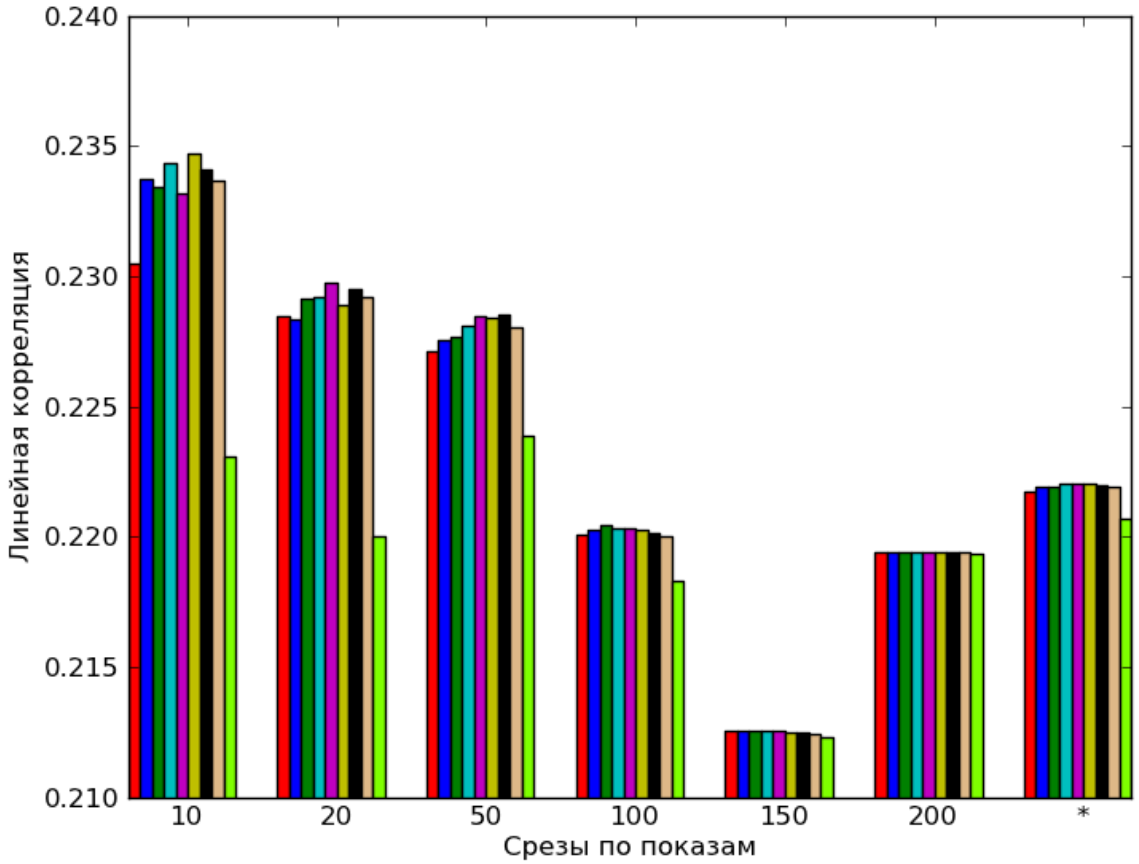
На рисунке 4 представлены полные результаты эксперимента. По вертикальной оси отложена линейная корреляция. По горизонтальной отложены различные срезы по показам. Столбцы в срезе соответствуют фильтрам, которые отсеивали баннеры, с историей большей 10, 20, 30, 50, 75, 100, 150, 200 показов соответственно. Последний столбик соотвествует полной выборке.

8 Результаты «on-line» эксперимента

Полученная формула была встроена в баннерную систему компании «Яндекс» и использовалась на некоторой доле запросов к поисковой системе. Результаты внедрения формулы со стартовым прогнозом были сравнены с текущей формулы по четырем критериям. Первый критерий — это средний CTR системы, т.е. отношение количества кликов по баннерам к количеству показов. Второй критерий — это абсолютное количество событий, соотвествующих показу одного баннера, нормированное на объем траффика. Третий — количество кликов, нормированное на объем траффика, а четвертый — средняя стоимость клика для рекламодателей. При этом сравнение старой и новой формул производилось не на всех баннерах, а только на новых. Сравнение проводено по двум срезам: для баннеров с количеством показов не превышающем 10 и для баннеров, имеющих от 10 до 100 показов. Результаты приведены в таблице 7.

На срезе, который соответствует самым новым баннерам CTR вырос на 22.7%, вместе с небольшим увеличением показов таких баннеров и значимым увеличением числа кликов на 26.3%. При этом цена клика упала на 26.6%. Результаты на этом срезе полностью соответствуют

Рис. 4. Результаты эксперимента с фильтрацией выборки



изначальной цели работы: новая реклама показывается более эффективно, она собирает больше кликов и стоит дешевле для рекламодателей.

На втором срезе видно, что конечные цели тоже достигнуты (CTR растет, средняя цена за клик падает), хотя общая картина немного другая. В этом случае количество показов, соответствующих баннерам с историей от 10 до 50 кликов уменьшилось на 16.2%. Это связано с тем, что старая формула делала более оптимистичный прогноз CTR для таких баннеров, чем новая.

На основе этих данных эксперимент был признан успешным и планируется дальнейшее внедрение формулы в баннерную систему.

9 Выводы

В ходе выполнения данной работы была решена задача предсказания стартового CTR новых баннеров. В отличие от предыдущих работ по этой теме в качестве целевой функции использовались непосредственно клики и неклики по баннерам, вместо их CTR. Также были исследованы

Таблица 7. Результаты «on-line» эксперимента

Длина истории	CTR	Показы	Клики	Цена клика
Не более 10 показов	+22.7%	+2.9%	+26.3%	-26.6%
От 10 до 50 показов	+21.0%	-16.2%	+1.4%	-7.3%

новые признаки, основанные на текстовой близости баннеров.

В итоге на основе построенной модели предсказания был проведен «on-line» эксперимент, результатом которого стало увеличение среднего CTR новых баннеров и уменьшение средней стоимости кликов по таким баннерам.

10 Результаты, выносимые на защиту

- Разработана и реализована модель предсказания вероятности клика (CTR) на новые баннеры, основанная на композиции решающих деревьев.
- Найдена система информативных признаков, экспериментально показана её безызбыточность.
- Модель внедрена в систему показов контекстной рекламы компании «Яндекс» и показала значительное улучшение качества прогноза CTR новых баннеров в реальном времени.

Список литературы

[1] Leo Breiman et al. *Classification and Regression Trees*. Chapman & Hall, 1984.

[2] JH Friedman. Greedy function approximation: A gradient boosting machine, 1999.

[3] C.D. Manning, P. Raghavan, and H. Schütze. Introduction to information retrieval. 2008.

[4] M. Regelson and D. Fain. Predicting click-through rate using keyword clusters. In *Proceedings of the Second Workshop on Sponsored Search Auctions*, volume 9623. Citeseer, 2006.

[5] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th international conference on World Wide Web*, pages 521–530. ACM, 2007.

[6] D. Smiley and E. Pugh. *Solr 1.4 Enterprise Search Server*. 2009.