Learning in Big Data Analytics
Lecture 4

Alexander Schönhuth

UNIVERSITÄT BIELEFELD
Faculty of Technology

Bielefeld University

December 8, 2020

Recap

- Placing web advertisements means assigning ads to search queries
  - Advertisers bid on queries
  - Advertisers have overall budget
  - Ads have click-through rate
- Ads need to be ranked according to bid, budget, rate to maximize revenue for search engine
- Decision need to be taken online, without delay
  ☞ Online algorithms
- *Competitive ratio* is fraction of revenue acquired with online relative to optimum offline algorithm
- Ads need to be matched with queries
  ☞ Matching algorithms
- Online matching well covered by *greedy algorithms*
- We computed the competitive ratio of greedy matching

The Adwords Problem

Search Advertizing Principle

Strategy by Overture [2000]

- Overture was company later acquired by Yahoo!
- Advertisers bid on keywords, as appearing in search queries
- *All* advertisers' links are displayed as response to user who searches keyword, highest-bid first order,
- Advertiser pays if links are clicked on
- Rather useless for users looking primarily for information
  ☞ which are the majority!
- Google adapted idea in system called *Adwords*
- Advertisers' links displayed separately from generic links

ADWORDS SYSTEM

Improvements

- Google displayed only limited list of advertisements: requires to decide which to show
- Advertisers have to specify an overall budget, the amount of money to spend for clicked-on ads in a given time (e.g. a month)
  ☞ more involved algorithmic problem
- Google evaluated click-through rates for ads to maximize profit

The Adwords Problem: Definition

Given

- Set of bids of advertisers for search queries
- Click-through rates for advertiser-query pairs
- Budget for each advertiser (usually specified for a month)
- Limit on number of ads to be displayed

Response to Search Query

- Set of ads no larger than the limit
- Each advertiser in the set has bid on query
- Each advertiser has sufficient budget left to pay bid

THE ADWORDS PROBLEM: DEFINITION

Adwords Algorithm: Target Function

- *Value* of ad is product of bid and click-through rate
- *Revenue* of selection of ads is sum of values
- *Merit* of an online-algorithm for determining selections of ads is revenue obtained over a month
- *Competitive ratio* is minimum of revenue for sequence of queries divided by revenue obtained for same sequence by optimum offline algorithm

Adwords Problem: Greedy Approach

Simplified Scenario

(a) One ad is shown for each query
(b) All advertisers have the same budget
(c) All click-through rates are the same
(d) All bids are 0 or 1

Alternative formulation of (d): the value (product bid times click-through rate) is the same for each advertiser.

Greedy Algorithm
For each search query, pick arbitrary advertiser

- who bids 1 on query
- has budget left

Adwords Problem: Note on Reality

Matching Bids with Search Queries

- Advertisers bid on sets of words
- *Exact matching:* eligible when query matches set of words exactly
- *Broad matching:* eligible also for inexact matches
  - Super- or subsets of words
  - Words that have similar meaning
  - Charging advertisers follows complicated formulas

Charging Advertisers for Clicks

- *First price auction:* Advertiser is charged the amount they bid
- *Second price auction:* Pay (approximate) bid of second placed advertiser
- Second price auctions less susceptible to being gamed by advertisers
  ☞ lead to higher revenues for search engines

Example

- Two advertisers, A_1 and A_2, each with budget 2
- Two possible queries, x and y; A_1 bids only on x, A_2 on x and y
- Consider sequence of queries *xxyy*
- The *Greedy algorithm*
  - can allocate the two x to A_2
  - A_1 does not bid on y, A_2 has no budget left
  - Revenue is 2
- The *Offline algorithm*
  - allocates the two x to A_1, and the two y to A_2
  - Revenue is 4
- The *competitive ratio* is thus no more than $\frac{2}{4} = \frac{1}{2}$.

The Balance Algorithm

Balance Algorithm

- Slight adaptation of Greedy algorithm
- Assigns query to advertiser who
  - bids on the query
  - *has the largest remaining budget*
  - Ties are broken arbitrarily

Example Revisited

Situation

- Two advertisers, A_1 and A_2, each with budget 2
- Two possible queries, x and y; A_1 bids only on x, A_2 on x and y
- Consider sequence of queries $xxyy$

Balance Algorithm

- Can put first x to A_2
- But then must put the second x to A_1
- Puts first y to A_2
- A_2 has no budget left to serve second y
- *Revenue* is 3, so *competitive ratio* is no more than $\frac{3}{4}$

BALANCE: LOWER BOUND COMPETITIVE RATIO

Situation

- Known upper bound on competitive ratio: $\frac{3}{4}$.
- Lower bound not known
- *Idea:* Establish a suitable lower bound

CLAIM

(i) A *lower bound* for the Balance algorithm, in the simple situation sketched (involving only 2 advertisers), is $\frac{3}{4}$

(ii) This establishes $\frac{3}{4}$ as the *competitive ratio* of the Balance algorithm

Note that (*ii*) is an immediate consequence of (*i*), when combining it with the upper bound we established.

Balance: Lower Bound Competitive Ratio II

Situation

- Two advertisers, A_1 and A_2, each of which has budget B
- *We need to show* that for an arbitrary sequence of queries, Balance achieves at least $\frac{3}{4}$ times the revenue of the optimum offline algorithm

Immediately Possible Assumptions

(*) Given two sequences of queries, we can focus on the sequence that provably yields a smaller ratio

☞ Suffices to show that the smaller ratio is at least $\frac{3}{4}$

(**) The optimum offline algorithm assigns each query to one of A_1 or A_2

☞ One can imagine to delete other queries without affecting the revenue, while the revenue of Balance can only decrease

- This yields a sequence whose ratio is smaller, make use of (*)

Balance: Lower Bound Competitive Ratio III

Situation

- Two advertisers, A_1 and A_2, each of which has budget B
- *We need to show* that for an arbitrary sequence of queries, Balance achieves at least $\frac{3}{4}$ times the revenue of the optimum offline algorithm

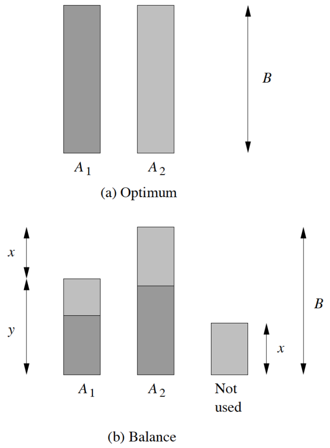
Immediately Possible Assumptions

(***) Both budgets are consumed by optimum offline algorithm

- If not, consider reduced, but fully consumed budgets
- Revenue of optimum offline algorithm remains the same
  - Note that the assumption of equal budget needs to be skipped
  - Ratio also applies for unequal budgets ☞ exercise!
- Balance revenue can only decrease

☞ Lowers ratio

Balance: Lower Bound Competitive Ratio IV

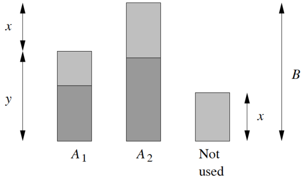


Adopted from `mmds.org`

- By assumption (***), the optimum algorithm consumes all budget 2B
- *Upper part* of image reflects necessary consequence

- One of the budgets must be fully consumed by Balance
- If not, query would be assigned to neither A_1, A_2, contradicting (**)
- *Lower part* reflects that A_2's budget is fully consumed

Balance: Lower Bound Competitive Ratio V



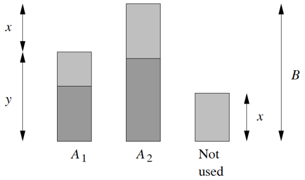
Adopted from `mmds.org`

- Some queries assigned to A_2 by Balance could have been assigned to A_1 by offline optimum (dark queries)
- Let y be number of queries assigned to A_1 (by Balance)
- Let $x = B - y$ be number of unassigned queries

We seek to show that

$$y \geq x \quad \text{implying that} \quad y \geq \frac{1}{2}B, \quad \text{yielding} \quad B + y \geq B + \frac{1}{2}B = \frac{3}{2}B \quad (1)$$

Balance: Lower Bound Competitive Ratio VI

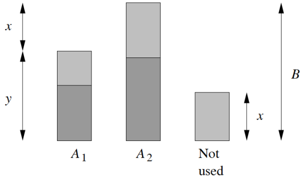


Adopted from `mmds.org`

- x is also the number of queries left unassigned by Balance
- All x queries must have gone to A_2 by the optimum algorithm
  - Assigning any of the x queries to A_1 means that A_1 would have bid on the queries
  - So, because A_1 had budget left, they would have been assigned to A_1 also by Balance

UNIVERSITÄT BIELEFELD

BALANCE: LOWER BOUND COMPETITIVE RATIO VI



Adopted from `mmds.org`

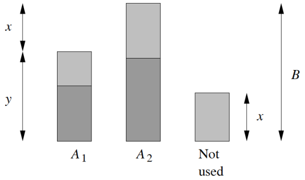
- Consider queries that are assigned to A_1 by Optimum (dark in figure)
- Recall that all such queries are assigned by Balance, either to A_1 or A_2

Two Cases

(i) More than half of dark queries are assigned to A_1 by Balance

(ii) More than half of dark queries are assigned to A_2 by Balance

Balance: Lower Bound Competitive Ratio VII



Adopted from `mmds.org`

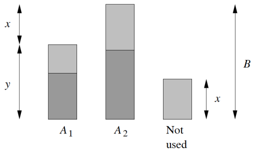
Two Cases

(i) More than half of dark queries are assigned to A_1 by Balance

(ii) More than half of dark queries are assigned to A_2 by Balance

Case (i): This case immediately implies that $y \geq B/2$, which implies $y \geq x$, so we are done.

Balance Algorithm: Lower Bound Competitive Ratio VI



Adopted from `mmds.org`

Case (ii): More than half of dark queries are assigned to A_2.

Consider the last dark query assigned to A_2 by Balance. At that point, A_2's budget must have been at least as great as A_1's budget, because otherwise, by the algorithmic principle of Balance, q would have been assigned to A_1 (+).

Since more than B/2 dark queries are assigned to A_2, A_2's budget was at most B/2 just before q arrived.

Because of (+), this implies that also A_1's budget was at most B/2, so A_1 had already collected at least B/2 queries. So $y \geq B/2$, implying $y \geq x$. □

Balance Algorithm with Many Bidders

The competitive ratio involving many bidders can be lower than $\frac{3}{4}$, but not much lower.

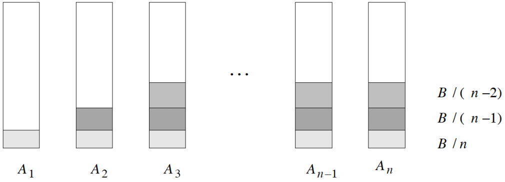
Worst-Case Scenario

1. There are N advertisers $A_1, ..., A_N$
2. Each advertiser has budget $B = N$!
3. There are N queries $q_1, ..., q_N$
4. Advertiser A_i bids on queries $q_1, ..., q_i$
5. The query sequence consists of N rounds, where the i-th round consists of B occurrences of q_i

Optimum Offline Algorithm

- Assigns all bids of i-th round to advertiser A_i
- Yields revenue $N \cdot B$

BALANCE ALGORITHM WITH MANY BIDDERS



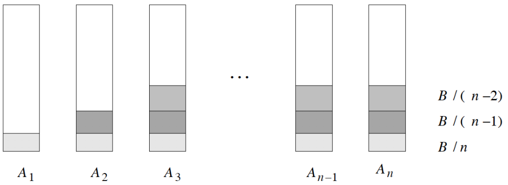
Adopted from `mmds.org`

Balance Algorithm

- Assigns all B occurrences of q_1 equally to all $A_i, i = 1, ..., N$
- Each advertiser gets B/N of queries q_1
- Assigns B occurrences of q_2 equally to all $A_i, i = 2, ..., n$
- Each of $A_2, ..., A_N$ gets $B/(N-1)$ of queries q_2
- . . .

UNIVERSITÄT BIELEFELD

BALANCE ALGORITHM WITH MANY BIDDERS

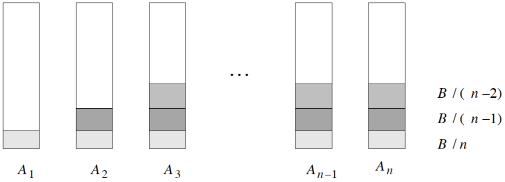


Adopted from `mmds.org`

Balance Algorithm

- ...
- $A_1, ..., A_N$ get $B/(N - i + 1)$ of queries q_i
- ...
- Eventually, budgets of higher-numbered advertisers will be exhausted

Balance Algorithm with Many Bidders



Adopted from `mmds.org`

Balance Algorithm

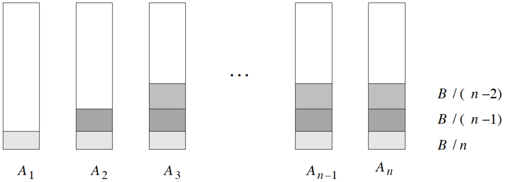
- Eventually, budgets of higher-numbered advertisers will be exhausted
- This happens at lowest round j where

$$B(\frac{1}{N} + \frac{1}{N-1} + ... + \frac{1}{N-j+1}) \geq B \quad (2)$$

that is, when

$$\frac{1}{N} + \frac{1}{N-1} + ... + \frac{1}{N-j+1} \geq 1 \quad (3)$$

BALANCE ALGORITHM WITH MANY BIDDERS



Adopted from `mmds.org`

Balance Algorithm

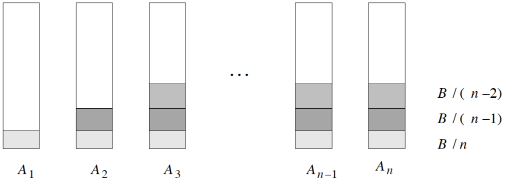
- Euler showed that

$$\sum_{i=1}^{k} \frac{1}{i} \xrightarrow{k\to\infty} \log_e k$$

- In other words, by approximating (3), we are looking for j where

$$\log_e N - \log_e(N-j) = 1 \quad \text{or, equivalently} \quad \frac{N}{N-j} = e \tag{4}$$

BALANCE ALGORITHM WITH MANY BIDDERS



Adopted from `mmds.org`

Balance Algorithm

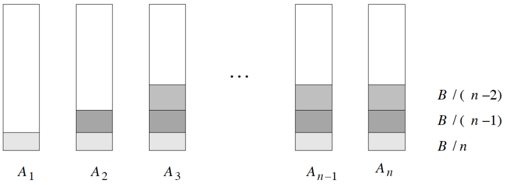
- In other words, by approximating (3), we are looking for j where

$$\log_e N - \log_e(N-j) = 1 \quad \text{or, equivalently} \quad \frac{N}{N-j} = e \qquad (5)$$

- Solving for j yields

$$j = N(1 - \frac{1}{e}) \qquad (6)$$

BALANCE ALGORITHM WITH MANY BIDDERS



Adopted from `mmds.org`

Balance Algorithm

- Solving for j yields $j = N(1 - \frac{1}{e})$
- So, the approximate revenue of Balance in this worst-case scenario is $BN(1 - \frac{1}{e})$
- This translates into a competitive ratio of

$$1 - \frac{1}{e} \approx 0.63$$

The Generalized Balance Algorithm

Situation
Advertisers' bids are arbitrary and not just 0 or 1

The following generalization of the Balance algorithm can be shown to have a competitive ratio of $1 - \frac{1}{e} \approx 0.63$:

Generalized Balance Algorithm

- Query q arrives
- Advertiser A_i has bid x_i for query q
- Advertiser A_i has fraction f_i of his budget left unspent
- Let

$$\Psi_i = x_i(1 - e^{-f_i}) \tag{7}$$

  Then assign q to advertiser A_i such that Ψ_i is maximum.

General / Further Reading

Literature

- Mining Massive Datasets, Section 8.4
  `http://infolab.stanford.edu/~ullman/mmds/ch8.pdf`