

## 32 Ranking

---

Ranking is the process of producing an ordered shortlist of  $K$  items from a larger collection of  $L$  items. These tasks come in several flavors. Sometimes the user supplies a query and the system responds with a shortlist of items. In other applications the shortlist is produced without an explicit query. For example, a streaming service might provide a list of recommended movies when you sign in. Our focus here is on the second type of problem.

We examine a sequential version of the ranking problem where the learner selects a ranking, receives feedback about its quality and repeats the process over  $n$  rounds. The feedback will be in the form of ‘clicks’ from the user, which comes from the view that ranking is a common application in on-line recommendation systems and the user selects the items they like by clicking on them. The objective of the learner is to maximize the expected number of clicks.

A **permutation** on  $[L]$  is an invertible function  $\sigma : [L] \rightarrow [L]$ . Let  $\mathcal{A}$  be the set of all permutations on  $[L]$ . In each round  $t$  the learner chooses an action  $A_t \in \mathcal{A}$ , which should be interpreted as meaning the learner places item  $A_t(k)$  in the  $k$ th position. Equivalently,  $A_t^{-1}(i)$  is the position of the  $i$ th item. Since the shortlist has length  $K$  the order of  $A_t(K+1), \dots, A_t(L)$  is not important and is included only for notational convenience. After choosing their action, the learner observes  $C_{ti} \in \{0, 1\}$  for each  $i \in [L]$  where  $C_{ti} = 1$  if the user clicked on the  $i$ th item in the collection. Note that the user may click on multiple items. We will assume a stochastic model where the probability that the user clicks on position  $k$  in round  $t$  only depends on  $A_t$  and is given by  $v(A_t, k)$  with  $v : \mathcal{A} \times [L] \rightarrow [0, 1]$  an unknown function. The regret over  $n$  rounds is

$$R_n = n \max_{a \in \mathcal{A}} \sum_{k=1}^L v(a, k) - \mathbb{E} \left[ \sum_{t=1}^n \sum_{i=1}^L C_{ti} \right].$$

A naive way to minimize the regret would be to create a finite-armed bandit where each arm corresponds to a ranking of the items and then apply your favourite algorithm from Part II. The problem is that these algorithms treat the arms as independent and cannot exploit any structure in the ranking. This is almost always unacceptable because the number of ways to rank  $K$  items from a collection of size  $L$  is  $L!/(L-K)!$ . Ranking illustrates one of the most fundamental dilemmas in machine learning: choosing a model. A rich model leads to low misspecification error, but takes longer to fit while a course model

can suffer from large misspecification error. In the context of ranking a model corresponds to assumptions on the function  $v$ .

## 32.1 Click models

The only way to avoid the curse of dimensionality is to make assumptions. A natural way to do this for ranking is to assume that the probability of clicking on an item depends on (a) the underlying quality of that item and (b) the location of that item in the chosen ranking. A formal definition of how this is done is called a **click model**. Deciding which model to use depends on the particulars of the problem at hand, such as how the list is presented to the user and whether or not clicking on an item diverts them to a different page. This issue has been studied by the data retrieval community and there is now a large literature devoted to the pros and cons of different choices. We limit ourselves to describing the popular choices and give pointers to the literature at the end of the chapter.

### *Document-based model*

The **document-based model** is one of the simplest click models, which assumes the probability of clicking on a shortlisted item is equal to its **attractiveness**. Formally, for each item  $i \in [L]$  let  $\alpha(i) \in [0, 1]$  be the attractiveness of item  $i$ . The document-based model assumes that

$$v(a, k) = \alpha(a(k))\mathbb{I}\{k \leq K\}.$$

The unknown quantity in this model is the attractiveness function, which has just  $L$  parameters.

### *Position-based model*

The document-based model might occasionally be justified, but in most cases the position of an item in the ranking also affects the likelihood of a click. A natural extension that accounts for this behavior is called the **position-based model**, which assumes that

$$v(a, k) = \alpha(a(k))\chi(k),$$

where  $\chi : [L] \rightarrow [0, 1]$  is a function that measures the quality of position  $k$ . Since the user cannot click on items that are not shown we assume that  $\chi(k) = 0$  for  $k > K$ . This model is richer than the document-based model, which is recovered by choosing  $\chi(k) = \mathbb{I}\{k \leq K\}$ . The number of parameters in the position-based models is  $K + L$ .

### *Cascade model*

The position-based model is not suitable for applications where clicking on an item takes the user to a different page. In the **cascade model** it is assumed that

the learner scans the shortlisted items in order and only clicks on the first item they find attractive. Define  $\chi : \mathcal{A} \times [L] \rightarrow [0, 1]$  by

$$\chi(a, k) = \begin{cases} 1 & \text{if } k = 1 \\ 0 & \text{if } k > K \\ \prod_{\ell=1}^{k-1} (1 - \alpha(a(\ell))) & \text{otherwise,} \end{cases}$$

which is the probability that the user has not clicked on the first  $k - 1$  items. Then the cascade model assumes that

$$v(a, k) = \alpha(a(k))\chi(a, k). \tag{32.1}$$

The first term in the factorization is the attractiveness function, which measures the probability that the user is attracted to the  $i$ th item. The second term can be interpreted as the probability that the user examines that item. This interpretation is also valid in the position-based model. It is important to emphasize that  $v(a, k)$  is the probability of clicking on the  $k$ th position when taking action  $a \in \mathcal{A}$ . This does not mean that  $C_{t_1}, \dots, C_{t_L}$  are independent. So far the assumptions only restrict the marginal distribution of each  $C_{ti}$ , which for most of this chapter is all we require. Nevertheless, in the cascade model it would be standard to assume that  $C_{t_{A_t}(k)} = 0$  if there exists an  $\ell < k$  such that  $C_{t_{A_t}(\ell)} = 1$  and otherwise

$$\mathbb{P}(C_{t_{A_t}(k)} = 1 \mid A_t, C_{t_{A_t}(1)} = 0, \dots, C_{t_{A_t}(k-1)} = 0) = \mathbb{I}\{k \leq K\} \alpha(A_t(k)).$$

Like the document-based model, the cascade model has  $L$  parameters.

*Generic model*

We now introduce a model that generalizes the last three. Previous models essentially assumed that the probability of a click factorizes into an attractiveness probability and an examination probability. We deviate from this norm by making assumptions directly on the function  $v$ . Given  $\alpha : [L] \rightarrow [0, 1]$ , an action  $a$  is called  $\alpha$ -optimal if the shortlisted items are the  $K$  most attractive sorted by attractiveness:  $\alpha(a(k)) = \max_{k' > k} \alpha(a(k'))$  for all  $k \in [K]$ .

ASSUMPTION 32.1 There exists an attractiveness function  $\alpha : [L] \rightarrow [0, 1]$  such that the following four conditions are satisfied. Let  $a \in \mathcal{A}$  and  $i, j, k \in [L]$  be such that  $\alpha(i) \geq \alpha(j)$  and let  $\sigma$  be the permutation that exchanges  $i$  and  $j$ .

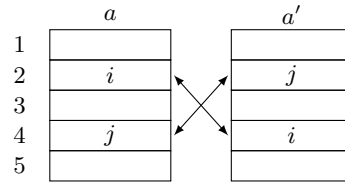
- (a)  $v(a, k) = 0$  for all  $k > K$ .
- (b)  $\sum_{k=1}^K v(a^*, k) = \max_{a \in \mathcal{A}} \sum_{k=1}^K v(a, k)$  for all  $\alpha$ -optimal actions  $a^*$ .
- (c) For all  $i$  and  $j$  with  $\alpha(i) \geq \alpha(j)$

$$v(a, a^{-1}(i)) \geq \frac{\alpha(i)}{\alpha(j)} v(\sigma \circ a, a^{-1}(i)),$$

where  $\sigma$  is the permutation on  $[L]$  that exchanges  $i$  and  $j$ .

- (d) If  $a$  is an action such that  $\alpha(a(k)) = \alpha(a^*(k))$  for some  $\alpha$ -optimal action  $a^*$ , then  $v(a, k) \geq v(a^*, k)$ .

These assumptions may appear quite mysterious. At some level they are chosen to make the proof go through, while simultaneously generalizing the document-based, position-based and cascade models (32.1). The choices not entirely without basis or intuition, however. Part (a) asserts that the user does not click on items that are not placed in the shortlist. Part (b) says that  $\alpha$ -optimal actions maximize the expected number of clicks. Note that there are multiple optimal rankings if  $\alpha$  is not injective. Part (c) is a little more restrictive and is illustrated in the figure. The probability of clicking on the second position is larger in the ranking on the left than the right by a factor of at least  $\alpha(i)/\alpha(j)$ . On the other hand, the probability of clicking on the fourth position is larger in the ranking on the right. One way to justify this is to assume that  $v(a, k) = \alpha(a(k))\chi(a, k)$  where  $\chi(a, k)$  is viewed as the probability that the user examines position  $k$ . It seems reasonable to assume that the probability the user examines position  $k$  should only depend on the first  $k-1$  items. Hence  $v(a, 2) = \alpha(i)\chi(a, 2) = \alpha(i)\chi(a', 2) = \alpha(i)/\alpha(j)v(a', 2)$ . In order to make the argument for the fourth position we need to assume that placing less attractive items in the early slots increases the probability that the user examines later positions (searching for a good result). This is true for the position-based and cascade models, but is perhaps the most easily criticised assumption. Part (d) says that the probability that a user clicks on a position with a correctly placed item is at least as large as the probability that the user clicks on that position in an optimal ranking. The justification is that the items  $a(1), \dots, a(k-1)$  cannot be more attractive than  $a^*(1), \dots, a^*(k-1)$ , which should increase the likelihood that the user makes it the  $k$ th position.



The generic model has many parameters, but we will see that the learner does not need to learn all of them in order to suffer small regret. The advantage of this model relative to the previous ones is that it offers more flexibility and yet it is not so flexible that learning is impossible.

## 32.2 Policy

We now explain the policy for learning to rank when  $v$  is unknown, but satisfies Assumption 32.1. After the description is an illustration that may prove helpful.

### Step 0: Initialization

The policy takes as input a confidence parameter  $\delta \in (0, 1)$  and  $L$  and  $K$ . The policy maintains a binary relation  $G_t \subseteq [L] \times [L]$ . In the first round  $t = 1$  the relation is empty:  $G_1 = \emptyset$ . You should think of  $G_t$  as maintaining pairs  $(i, j)$  for which the policy has proven with high probability that  $\alpha(i) < \alpha(j)$ . Ideally,  $G_t \subseteq G = \{(i, j) \in [L] \times [L] : \alpha(i) \leq \alpha(j)\}$ .

*Step 1: Defining a partition*

In each round  $t$  the learner computes a partition of the actions based on a topological sort according to relation  $G_t$ . Given  $A \subset [L]$  define  $\min_{G_t}(A)$  to be the set of minimum elements of  $A$  according to relation  $G_t$ .

$$\min_{G_t}(A) = \{i \in A : (i, j) \notin G_t \text{ for all } j \in A\}.$$

Then let  $\mathcal{P}_{t1}, \mathcal{P}_{t2}, \dots$  be the partition of  $[L]$  defined inductively by

$$\mathcal{P}_{td} = \min_{G_t} \left( [L] \setminus \bigcup_{c=1}^{d-1} \mathcal{P}_{tc} \right).$$

Finally, let  $M_t = \max\{d : \mathcal{P}_{td} \neq \emptyset\}$ . The reader should check that if  $G_t$  does not have cycles, then  $M_t$  is well defined and finite and that  $\mathcal{P}_{t1}, \dots, \mathcal{P}_{tM_t}$  is indeed a partition of  $[L]$  (Exercise 32.5). The event that  $G_t$  contains cycles is a failure event. In order for the policy to be well defined we assume it chooses some arbitrary fixed action in this case.

*Step 2: Choosing an action*

Define  $\mathcal{I}_{t1}, \dots, \mathcal{I}_{tM_t}$  be the partition of  $[L]$  defined inductively by

$$\mathcal{I}_{td} = [ \bigcup_{c \leq d} \mathcal{P}_{tc} ] \setminus [ \bigcup_{c < d} \mathcal{P}_{tc} ].$$

Next let  $\Sigma_t \subseteq \mathcal{A}$  be the set of actions  $\sigma$  such that  $\sigma(\mathcal{I}_{td}) = \mathcal{P}_{td}$  for all  $d \in [M_t]$ . The algorithm chooses  $A_t$  uniformly at random from  $\Sigma_t$ . Intuitively the policy first shuffles the items in  $\mathcal{P}_{t1}$  and uses these as the first  $|\mathcal{P}_{t1}|$  entries in the ranking. Then  $\mathcal{P}_{t2}$  is shuffled and the items are appended to the ranking. This process is repeated until the ranking is complete. For an item  $i \in [L]$ , we denote by  $D_{ti}$  the unique index  $d$  such that  $i \in \mathcal{P}_{td}$ .

*Step 3: Updating the relation*

For any pair of items  $i, j \in [L]$  define  $S_{tij} = \sum_{s=1}^t U_{sij}$  and  $N_{tij} = \sum_{s=1}^t |U_{sij}|$  where

$$U_{tij} = \mathbb{I}\{D_{ti} = D_{tj}\} (C_{ti} - C_{tj}).$$

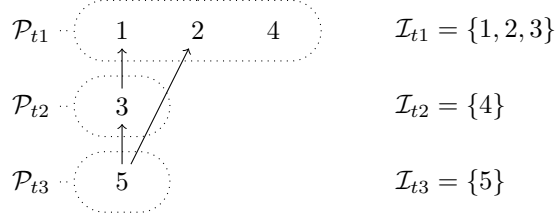
All this means is that  $S_{tij}$  tracks the differences between the number of clicks of items  $i$  and  $j$  over rounds when they share a partition. As a final step, the relation  $G_{t+1}$  is given by

$$G_{t+1} = G_t \cup \left\{ (j, i) : S_{tij} \geq \sqrt{2N_{tij} \log \left( \frac{c\sqrt{N_{tij}}}{\delta} \right)} \right\},$$


where  $c \approx 3.43$  is the universal constant given in Exercise 20.7. In the analysis we will show that if  $\alpha(i) \geq \alpha(j)$ , then with high probability  $S_{tji}$  is never large enough for  $G_{t+1}$  to include  $(i, j)$ . In this sense, with high probability  $G_t$  is consistent with the order on  $[L]$  induced by sorting in decreasing order with respect to  $\alpha(\cdot)$ . Note that  $G_t$  is generally not a partial order because it is not transitive.

*Illustration*

Suppose  $L = 5$  and  $K = 4$  and in round  $t$  the relation is  $G_t = \{(3, 1), (5, 2), (5, 3)\}$ , which is represented in the graph below where an arrow from  $j$  to  $i$  indicates that  $(j, i) \in G_t$ .



This means that in round  $t$  the first three positions in the ranking will contain items from  $\mathcal{P}_{t1} = \{1, 2, 4\}$ , but with random order. The fourth position will be item 3 and item 5 is not shown to the user.

 Part (a) of Assumption 32.1 means that items in position  $k > K$  are never clicked. As a consequence, the algorithm never needs to actually compute the partitions  $\mathcal{P}_{td}$  for which  $\min \mathcal{I}_{td} > K$  because items in these partitions are never shortlisted.

### 32.3 Regret analysis

**THEOREM 32.1** *Let function  $v$  satisfy Assumption 32.1 and assume that  $\alpha(1) > \alpha(2) > \dots > \alpha(L)$ . Let  $\Delta_{ij} = \alpha(i) - \alpha(j)$  and  $\delta \in (0, 1)$ . Then the regret of TOPRANK is bounded from above as*

$$R_n \leq \delta n K L^2 + \sum_{j=1}^L \sum_{i=1}^{\min\{K, j-1\}} \left( 1 + \frac{6(\alpha(i) + \alpha(j)) \log\left(\frac{c\sqrt{n}}{\delta}\right)}{\Delta_{ij}} \right).$$

Furthermore,  $R_n \leq \delta n K L^2 + K L + \sqrt{4K^3 L n \log\left(\frac{c\sqrt{n}}{\delta}\right)}.$

By choosing  $\delta = n^{-1}$  the theorem shows that the expected regret is at most

$$R_n = O\left(\sum_{j=1}^L \sum_{i=1}^{\min\{K, j-1\}} \frac{\alpha(i) \log(n)}{\Delta_{ij}}\right) \quad \text{and} \quad R_n = O\left(\sqrt{K^3 L n \log(n)}\right).$$

The algorithm does not make use of any assumed ordering on  $\alpha(\cdot)$ , so the assumption is only used to allow for a simple expression for the regret. The core idea of the proof is to show that (a) if the algorithm is suffering regret as a consequence of misplacing an item, then it is gaining information about the relation of the items so that  $G_t$  will gain elements and (b) once  $G_t$  is sufficiently

rich the algorithm is playing optimally. Let  $\mathcal{F}_t = \sigma(A_1, C_1, \dots, A_t, C_t)$  and  $\mathbb{P}_t(\cdot) = \mathbb{P}(\cdot \mid \mathcal{F}_t)$  and  $\mathbb{E}_t[\cdot] = \mathbb{E}[\cdot \mid \mathcal{F}_t]$ . For each  $t \in [n]$  let  $F_t$  to be the failure event that there exists  $i \neq j \in [L]$  and  $s < t$  such that  $N_{sij} > 0$  and

$$\left| S_{sij} - \sum_{u=1}^s \mathbb{E}_{u-1}[U_{uij} \mid U_{uij} \neq 0] \mid U_{uij} \right| \geq \sqrt{2N_{sij} \log(c\sqrt{N_{sij}}/\delta)}.$$

**LEMMA 32.1** *Let  $i$  and  $j$  satisfy  $\alpha(i) \geq \alpha(j)$  and  $d \geq 1$ . On the event that  $i, j \in \mathcal{P}_{sd}$  and  $d \in [M_s]$  and  $U_{sij} \neq 0$ , the following hold almost surely:*

$$(a) \mathbb{E}_{s-1}[U_{sij} \mid U_{sij} \neq 0] \geq \frac{\Delta_{ij}}{\alpha(i) + \alpha(j)}.$$

$$(b) \mathbb{E}_{s-1}[U_{sji} \mid U_{sji} \neq 0] \leq 0.$$

*Proof* For the remainder of the proof we focus on the event that  $i, j \in \mathcal{P}_{sd}$  and  $d \in [M_s]$  and  $U_{sij} \neq 0$ . We also discard the measure zero subset of this event where  $\mathbb{P}_{s-1}(U_{sij} \neq 0) = 0$ . From now on we omit the ‘almost surely’ qualification on conditional expectations. Under these circumstances the definition of conditional expectation shows that

$$\begin{aligned} \mathbb{E}_{s-1}[U_{sij} \mid U_{sij} \neq 0] &= \frac{\mathbb{P}_{s-1}(C_{si} = 1, C_{sj} = 0) - \mathbb{P}_{s-1}(C_{si} = 0, C_{sj} = 1)}{\mathbb{P}_{s-1}(C_{si} \neq C_{sj})} \\ &= \frac{\mathbb{P}_{s-1}(C_{si} = 1) - \mathbb{P}_{s-1}(C_{sj} = 1)}{\mathbb{P}_{s-1}(C_{si} \neq C_{sj})} \\ &\geq \frac{\mathbb{P}_{s-1}(C_{si} = 1) - \mathbb{P}_{s-1}(C_{sj} = 1)}{\mathbb{P}_{s-1}(C_{si} = 1) + \mathbb{P}_{s-1}(C_{sj} = 1)} \\ &= \frac{\mathbb{E}_{s-1}[v(A_s, A_s^{-1}(i)) - v(A_s, A_s^{-1}(j))]}{\mathbb{E}_{s-1}[v(A_s, A_s^{-1}(i)) + v(A_s, A_s^{-1}(j))]}, \end{aligned} \quad (32.2)$$

where in the second equality we added and subtracted  $\mathbb{P}_{s-1}(C_{si} = 1, C_{sj} = 1)$ . By the design of TOPRANK, the items in  $\mathcal{P}_{td}$  are placed into slots  $\mathcal{I}_{td}$  uniformly at random. Let  $\sigma$  be the permutation that exchanges the positions of items  $i$  and  $j$ . Then using Part Item (c) of Assumption 32.1,

$$\begin{aligned} \mathbb{E}_{s-1}[v(A_s, A_s^{-1}(i))] &= \sum_{a \in \mathcal{A}} \mathbb{P}_{s-1}(A_s = a) v(a, a^{-1}(i)) \\ &\geq \frac{\alpha(i)}{\alpha(j)} \sum_{a \in \mathcal{A}} \mathbb{P}_{s-1}(A_s = a) v(\sigma \circ a, a^{-1}(i)) \\ &= \frac{\alpha(i)}{\alpha(j)} \sum_{a \in \mathcal{A}} \mathbb{P}_{s-1}(A_s = \sigma \circ a) v(\sigma \circ a, (\sigma \circ a)^{-1}(j)) \\ &= \frac{\alpha(i)}{\alpha(j)} \mathbb{E}_{s-1}[v(A_s, A_s^{-1}(j))], \end{aligned}$$

where the second equality follows from the fact that  $a^{-1}(i) = (\sigma \circ a)^{-1}(j)$  and the definition of the algorithm ensuring that  $\mathbb{P}_{s-1}(A_s = a) = \mathbb{P}_{s-1}(A_s = \sigma \circ a)$ . The

last equality follows from the fact that  $\sigma$  is a bijection. Using this and continuing the calculation in Eq. (32.2) shows that

$$\begin{aligned} \text{Eq. (32.2)} &= \frac{\mathbb{E}_{s-1}[v(A_s, A_s^{-1}(i)) - v(A_s, A_s^{-1}(j))]}{\mathbb{E}_{s-1}[v(A_s, A_s^{-1}(i)) + v(A_s, A_s^{-1}(j))]} \\ &= 1 - \frac{2}{1 + \mathbb{E}_{s-1}[v(A_s, A_s^{-1}(i))] / \mathbb{E}_{s-1}[v(A_s, A_s^{-1}(j))]} \\ &\geq 1 - \frac{2}{1 + \alpha(i)/\alpha(j)} \\ &= \frac{\alpha(i) - \alpha(j)}{\alpha(i) + \alpha(j)} = \frac{\Delta_{ij}}{\alpha(i) + \alpha(j)}. \end{aligned}$$

The second part follows from the first since  $U_{sji} = -U_{sij}$ .  $\square$

The next lemma shows that the failure event occurs with low probability.

LEMMA 32.2 *It holds that  $\mathbb{P}(F_n) \leq \delta L^2$ .*

*Proof* The proof follows immediately from Lemma 32.1, the definition of  $F_n$ , the union bound over all pairs of actions, and a modification of the Azuma-Hoeffding inequality in Exercise 20.7.  $\square$

LEMMA 32.3 *On the event  $F_t^c$  it holds that  $(i, j) \notin G_t$  for all  $i < j$ .*

*Proof* Let  $i < j$  so that  $\alpha(i) \geq \alpha(j)$ . On the event  $F_t^c$  either  $N_{sji} = 0$  or

$$S_{sji} - \sum_{u=1}^s \mathbb{E}_{u-1}[U_{uji} \mid U_{uji} \neq 0] |U_{uji}| < \sqrt{2N_{sji} \log\left(\frac{c}{\delta} \sqrt{N_{sji}}\right)} \quad \text{for all } s < t.$$

When  $i$  and  $j$  are in different blocks in round  $u < t$ , then  $U_{uji} = 0$  by definition. On the other hand, when  $i$  and  $j$  are in the same block,  $\mathbb{E}_{u-1}[U_{uji} \mid U_{uji} \neq 0] \leq 0$  almost surely by Lemma 32.1. Based on these observations,

$$S_{sji} < \sqrt{2N_{sji} \log\left(\frac{c}{\delta} \sqrt{N_{sji}}\right)} \quad \text{for all } s < t,$$

which by the design of TOPRANK implies that  $(i, j) \notin G_t$ .  $\square$

LEMMA 32.4 *Let  $I_{td}^* = \min \mathcal{P}_{td}$  be the most attractive item in  $\mathcal{P}_{td}$ . Then on event  $F_t^c$ , it holds that  $I_{td}^* \leq 1 + \sum_{c < d} |\mathcal{P}_{td}|$  for all  $d \in [M_t]$ .*

*Proof* Let  $i^* = \min \cup_{c \geq d} \mathcal{P}_{tc}$ . Then  $i^* \leq 1 + \sum_{c < d} |\mathcal{P}_{td}|$  holds trivially for any  $\mathcal{P}_{t1}, \dots, \mathcal{P}_{tM_t}$  and  $d \in [M_t]$ . Now consider two cases. Suppose that  $i^* \in \mathcal{P}_{td}$ . Then it must be true that  $i^* = I_{td}^*$  and our claim holds. On other hand, suppose that  $i^* \in \mathcal{P}_{tc}$  for some  $c > d$ . Then by Lemma 32.3 and the design of the partition, there must exist a sequence of items  $i_d, \dots, i_c$  in blocks  $\mathcal{P}_{td}, \dots, \mathcal{P}_{tc}$  such that  $i_d < \dots < i_c = i^*$ . From the definition of  $I_{td}^*$ ,  $I_{td}^* \leq i_d < i^*$ . This concludes our proof.  $\square$



LEMMA 32.5 On the event  $F_n^c$  and for all  $i < j$  it holds that

$$S_{nij} \leq 1 + \frac{6(\alpha(i) + \alpha(j))}{\Delta_{ij}} \log \left( \frac{c\sqrt{n}}{\delta} \right).$$

*Proof* The result is trivial when  $N_{nij} = 0$ . Assume from now on that  $N_{nij} > 0$ . By the definition of the algorithm arms  $i$  and  $j$  are not in the same block once  $S_{tij}$  grows too large relative to  $N_{tij}$ , which means that

$$S_{nij} \leq 1 + \sqrt{2N_{nij} \log \left( \frac{c}{\delta} \sqrt{N_{nij}} \right)}.$$

On the event  $F_n^c$  and part (a) of Lemma 32.1 it also follows that

$$S_{nij} \geq \frac{\Delta_{ij} N_{nij}}{\alpha(i) + \alpha(j)} - \sqrt{2N_{nij} \log \left( \frac{c}{\delta} \sqrt{N_{nij}} \right)}.$$

Combining the previous two displays shows that

$$\begin{aligned} \frac{\Delta_{ij} N_{nij}}{\alpha(i) + \alpha(j)} - \sqrt{2N_{nij} \log \left( \frac{c}{\delta} \sqrt{N_{nij}} \right)} &\leq S_{nij} \leq 1 + \sqrt{2N_{nij} \log \left( \frac{c}{\delta} \sqrt{N_{nij}} \right)} \\ &\leq (1 + \sqrt{2}) \sqrt{N_{nij} \log \left( \frac{c}{\delta} \sqrt{N_{nij}} \right)}. \end{aligned} \quad (32.3)$$

Using the fact that  $N_{nij} \leq n$  and rearranging the terms in the previous display shows that

$$N_{nij} \leq \frac{(1 + 2\sqrt{2})^2 (\alpha(i) + \alpha(j))^2}{\Delta_{ij}^2} \log \left( \frac{c\sqrt{n}}{\delta} \right).$$

The result is completed by substituting this into Eq. (32.3).  $\square$

*Proof of Theorem 32.1* The first step in the proof is an upper bound on the expected number of clicks in the optimal list  $a^*$ . Fix time  $t$ , block  $\mathcal{P}_{td}$ , and recall that  $I_{td}^* = \min \mathcal{P}_{td}$  is the most attractive item in  $\mathcal{P}_{td}$ . Let  $k = A_t^{-1}(I_{td}^*)$  be the position of item  $I_{td}^*$  and  $\sigma$  be the permutation that exchanges items  $k$  and  $I_{td}^*$ . By Lemma 32.4,  $I_{td}^* \leq k$ ; and then from Parts (c) and (d) of Assumption 32.1 we have that  $v(A_t, k) \geq v(\sigma \circ A_t, k) \geq v(a^*, k)$ . Based on this result, the expected number of clicks on  $I_{td}^*$  is bounded from below by those on items in  $a^*$ ,

$$\begin{aligned} \mathbb{E}_{t-1} [C_{tI_{td}^*}] &= \sum_{k \in \mathcal{I}_{td}} \mathbb{P}_{t-1}(A_t^{-1}(I_{td}^*) = k) \mathbb{E}_{t-1}[v(A_t, k) \mid A_t^{-1}(I_{td}^*) = k] \\ &= \frac{1}{|\mathcal{I}_{td}|} \sum_{k \in \mathcal{I}_{td}} \mathbb{E}_{t-1}[v(A_t, k) \mid A_t^{-1}(I_{td}^*) = k] \geq \frac{1}{|\mathcal{I}_{td}|} \sum_{k \in \mathcal{I}_{td}} v(a^*, k), \end{aligned}$$

where we also used the fact that TOPRANK randomizes within each block to guarantee that  $\mathbb{P}_{t-1}(A_t^{-1}(I_{td}^*) = k) = 1/|\mathcal{I}_{td}|$  for any  $k \in \mathcal{I}_{td}$ . Using this and the design of TOPRANK,

$$\sum_{k=1}^K v(a^*, k) = \sum_{d=1}^{M_t} \sum_{k \in \mathcal{I}_{td}} v(a^*, k) \leq \sum_{d=1}^{M_t} |\mathcal{I}_{td}| \mathbb{E}_{t-1} [C_{tI_{td}^*}].$$

Therefore, under event  $F_t^c$ , the conditional expected regret in round  $t$  is bounded by

$$\begin{aligned}
 \sum_{k=1}^K v(a^*, k) - \mathbb{E}_{t-1} \left[ \sum_{j=1}^L C_{tj} \right] &\leq \mathbb{E}_{t-1} \left[ \sum_{d=1}^{M_t} |\mathcal{P}_{td}| C_{tI_{td}^*} - \sum_{j=1}^L C_{tj} \right] \\
 &= \mathbb{E}_{t-1} \left[ \sum_{d=1}^{M_t} \sum_{j \in \mathcal{P}_{td}} (C_{tI_{td}^*} - C_{tj}) \right] \\
 &= \sum_{d=1}^{M_t} \sum_{j \in \mathcal{P}_{td}} \mathbb{E}_{t-1} [U_{tI_{td}^*j}] \\
 &\leq \sum_{j=1}^L \sum_{i=1}^{\min\{K, j-1\}} \mathbb{E}_{t-1} [U_{tij}]. \tag{32.4}
 \end{aligned}$$

The last inequality follows by noting that  $\mathbb{E}_{t-1}[U_{tI_{td}^*j}] \leq \sum_{i=1}^{\min\{K, j-1\}} \mathbb{E}_{t-1}[U_{tij}]$ . To see this use part (a) of Lemma 32.1 to show that  $\mathbb{E}_{t-1}[U_{tij}] \geq 0$  for  $i < j$  and Lemma 32.4 to show that when  $I_{td}^* > K$ , then neither  $I_{td}^*$  nor  $j$  are not shown to the user in round  $t$  so that  $U_{tI_{td}^*j} = 0$ . Substituting the bound in Eq. (32.4) into the regret leads to

$$R_n \leq nK\mathbb{P}(F_n) + \sum_{j=1}^L \sum_{i=1}^{\min\{K, j-1\}} \mathbb{E} [\mathbb{I}\{F_n^c\} S_{nij}], \tag{32.5}$$

where we used the fact that the maximum number of clicks over  $n$  rounds is  $nK$ . The proof of the first part is completed by using Lemma 32.2 to bound the first term and Lemma 32.5 to bound the second. The problem independent bound follows from Eq. (32.5) and by stopping early in the proof of Lemma 32.5 (Exercise 32.6).  $\square$

## 32.4 Notes

- 1 At no point in the analysis did we use the fact that  $v$  is fixed over time. Suppose that  $v_1, \dots, v_n$  are a sequence of click-probability functions that all satisfy Assumption 32.1 with the same attractiveness function. The regret in this setting is

$$R_n = \sum_{t=1}^n \sum_{k=1}^K v_t(a^*, k) - \mathbb{E} \left[ \sum_{t=1}^n \sum_{i=1}^L C_{ti} \right].$$

- Then the bounds in Theorem 32.1 still hold without changing the algorithm.
- 2 The cascade model is usually formalized in the following more restrictive fashion. Let  $\{Z_{ti} : i \in [L], t \in [n]\}$  be a collection of independent Bernoulli random

variables with  $\mathbb{P}(Z_{ti} = 1) = \alpha(i)$ . Then define  $K_t$  as the first item  $i$  in the shortlist with  $Z_{ti} = 1$ :

$$K_t = \min \{k \in [K] : Z_{tA_t(k)} = 1\},$$

where the minimum of an empty set is  $\infty$ . Finally let  $C_{ti} = 1$  if and only if  $K_t \leq K$  and  $A_t(K_t) = i$ . This setup satisfies Eq. (32.1), but the independence assumption makes it possible to estimate  $\alpha$  without randomization. Notice that in any round  $t$  with  $K_t \leq K$ , all items  $i$  with  $A_t^{-1}(i) < K_t$  must have been unattractive ( $Z_{ti} = 0$ ) while the clicked item must be attractive ( $Z_{ti} = 1$ ). This fact can be used in combination with standard concentration analysis to estimate the attractiveness. The optimistic policy sorts the  $L$  items in decreasing order by their upper confidence bounds and shortlists the first  $K$ . When the confidence bounds are derived from Hoeffding’s inequality this policy is called CascadeUCB, while the policy that uses Chernoff’s lemma is called CascadeKL-UCB. The computational cost of the latter policy is marginally higher than the former, but the improvement is also quite significant because in practice most items have barely positive attractiveness.

- 3 The linear dependence of the regret on  $L$  is unpleasant when the number of items is large, which is the case in many practical problems. Like for finite-armed bandits one can introduce a linear structure on the items by assuming that  $\alpha(i) = \langle \theta, \phi_i \rangle$  where  $\theta \in \mathbb{R}^d$  is an unknown parameter vector and  $(\phi_i)_{i=1}^L$  are known feature vectors. This has been investigated in the cascade model by Zong et al. [2016].
- 4 There is an adversarial variant of the cascade model. In the **ranked bandit model** an adversary secretly chooses a sequence of sets  $S_1, \dots, S_n$  with  $S_t \subseteq [L]$ . In each round  $t$  the learner chooses  $A_t \in \mathcal{A}$  and receives a reward  $X_t(A_t)$  where  $X_t : \mathcal{A} \rightarrow [0, 1]$  is given by  $X_t(a) = \mathbb{I}\{S_t \cap \{a(1), \dots, a(k)\} \neq \emptyset\}$ . The feedback is the position of the clicked action, which is  $K_t = \min\{k \in [K] : A_t(k) \in S_t\}$ . The regret is

$$R_n = \sum_{t=1}^n (X_t(a_*) - X_t(A_t)),$$

where  $a_*$  is the optimal ranking in hindsight:

$$a_* = \operatorname{argmin}_{a \in \mathcal{A}} \sum_{t=1}^n X_t(a). \tag{32.6}$$

Notice that this is the same as the cascade model when  $S_t = \{i : Z_{ti} = 1\}$ .

- 5 A challenge in the ranked bandit model is that solving the offline problem (Eq. 32.6) for known  $S_1, \dots, S_n$  is NP-hard. How can one learn when finding an optimal solution to the offline problem is hard? First, hardness only matters if  $|\mathcal{A}|$  is large. When  $L$  and  $K$  are not too large, then exhaustive search is a quite feasible. If this is not an option one may use an approximation algorithm. It turns out that in a certain sense the best one can do is to use a greedy

algorithm, We omit the details, but the highlight is that there exist efficient algorithms such that

$$\mathbb{E} \left[ \sum_{t=1}^n X_t(A_t) \right] \geq \left(1 - \frac{1}{e}\right) \max_{a \in \mathcal{A}} \sum_{t=1}^n X_t(a) - O\left(K \sqrt{nL \log(L)}\right).$$

See the article by [Radlinski et al. \[2008\]](#) for more details.

- 6 By modifying the reward function one can also define an adversarial variant of the document-based model. Like before the adversary secretly chooses  $S_1, \dots, S_n$  as subsets of  $[L]$ , but now the reward is

$$X_t(a) = |S_t \cap \{a(1), \dots, a(k)\}|.$$

The feedback is the positions of the clicked items,  $S_t \cap \{a(1), \dots, a(k)\}$ . For this model there are no computation issues. In fact, problem can be analyzed using a reduction to combinatorial semibandits, which we ask you to investigate in [Exercise 32.3](#).

- 7 The position-based model can also be modelled in the adversarial setting by letting  $S_{tk} \subset [L]$  for each  $t \in [n]$  and  $k \in [K]$ . Then defining the reward by

$$X_t(a) = \sum_{k=1}^K \mathbb{I}\{A_t(k) \in S_{tk}\}.$$

Again, the feedback is the positions of the clicked items,  $\{k \in [K] : A_t(k) \in S_{tk}\}$ . This model can also be tackled using algorithms for combinatorial semibandits ([Exercise 32.4](#)).

## 32.5 Bibliographic remarks

The policy and analysis presented in this chapter is by the authors and others [[Lattimore et al., 2018](#)]. The most related work is by [Zoghi et al. \[2017\]](#) who assumed a factorization of the click probabilities  $v(a, k) = \alpha(a(k))\chi(a, k)$  and then made assumptions on  $\chi$ . The assumptions made here are slightly less restrictive and the bounds are simultaneously stronger. Some experimental results comparing these algorithms are given by [Lattimore et al. \[2018\]](#). For more information on click models we recommend the survey paper by [Chuklin et al. \[2015\]](#) and article by [Craswell et al. \[2008\]](#). Cascading bandits were first studied by [Kveton et al. \[2015a\]](#), who proposed algorithms based on UCB and KL-UCB and prove finite-time instance-dependence upper bounds and asymptotic lower bounds that match in specific regimes. Around the same time [Combes et al. \[2015\]](#) proposed a different algorithm for the same model that is also asymptotically optimal. The optimal regret has a complicated form and is not given explicitly in all generality. We remarked in the notes that the linear dependence on  $L$  is problematic for large  $L$ . To overcome this problem [Zong et al. \[2016\]](#) introduce a linear variant where the attractiveness of an item is assumed to be an inner product between an

unknown parameter and a known feature vector. A slightly generalized version of this setup was simultaneously studied by [Li et al. \[2016\]](#), who allowed the features associated with each item to change from round to round. The position-based model is studied by [Lagree et al. \[2016\]](#) who suggest several algorithms and provide logarithmic regret analysis for some of them. Asymptotic lower bounds are also given that match the upper bounds in some regimes. [Katariya et al. \[2016\]](#) study the **dependent click model** introduced by [Guo et al. \[2009\]](#). This differs from the models proposed in this chapter because the reward is not assumed to be the number of clicks and is actually unobserved. We leave the reader to explore this interesting model on their own. The adversarial variant of the ranking problem mentioned in the notes is due to [Radlinski et al. \[2008\]](#). Another related problem is the rank-1 bandit problem where the learner chooses one of  $L$  items to place in one of  $K$  positions, with all other positions left empty. This model has been investigated by [Katariya et al. \[2017b,a\]](#), who assume the position-based model. The cascade feedback model is also used in a combinatorial setting by [Kveton et al. \[2015c\]](#), but this paper does not have a direct application to ranking.

## 32.6 Exercises

**32.1** Show that the document-based, position-based and cascade models all satisfy Assumption 32.1.

**32.2** Most ranking algorithms are based on assigning an attractiveness value to each item and shortlisting the  $K$  most attractive items. [Radlinski et al. \[2008\]](#) criticize this approach in their paper as follows:

“The theoretical model that justifies ranking documents in this way is the probabilistic ranking principle [[Robertson, 1977](#)]. It suggests that documents should be ranked by their probability of relevance to the query. However, the optimality of such a ranking relies on the assumption that there are no statistical dependencies between the probabilities of relevance among documents – an assumption that is clearly violated in practice. For example, if one document about jaguar cars is not relevant to a user who issues the query jaguar, other car pages become less likely to be relevant. Furthermore, empirical studies have shown that given a fixed query, the same document can have different relevance to different users [[Teevan et al., 2007](#)]. This undermines the assumption that each document has a single relevance score that can be provided as training data to the learning algorithm. Finally, as users are usually satisfied with finding a small number of, or even just one, relevant document, the usefulness and relevance of a document does depend on other documents ranked higher.”

The optimality criterion [Radlinski et al. \[2008\]](#) had in mind is to present at least one item that the user is attracted to. Do you find this argument convincing? Why or why not?



The probabilistic ranking principle was put forward by [Maron and Kuhns \[1960\]](#). The paper by [Robertson \[1977\]](#) identifies some sufficient conditions under which the principle is valid and also discusses its limitations.

**32.3** Frame the adversarial variant of the document-based model in [Note 6](#) as a combinatorial semibandit and use the results in [Chapter 30](#) to prove a bound on the regret of

$$R_n \leq \sqrt{2KLn(1 + \log(L))}.$$

**32.4** Adapt your solution to the previous exercise to the position-based model in [Note 7](#) and prove a bound on the regret of

$$R_n \leq K\sqrt{2Ln(1 + \log(L))}.$$

**32.5** Prove that if  $G_t$  does not contain cycles, then  $M_t$  defined in [Section 32.2](#) is well defined and that  $\mathcal{P}_{t1}, \dots, \mathcal{P}_{tM_t}$  is a partition of  $[L]$ .

**32.6** Prove the second part of [Theorem 32.1](#).