
Ranking with unlabeled Data: A first study

Nicolas Usunier, Vinh Truong, Massih R. Amini, Patrick Gallinari
University Pierre and Marie Curie
Computer Science Department
75015 Paris, France
{usunier, truong, amini, gallinari}@poleia.lip6.fr

Abstract

In this paper, we present a general learning framework which treats the ranking problem for various Information Retrieval tasks. We extend the training set generalization error bound proposed by [4] to the ranking case and show that the use of unlabeled data can be beneficial for learning a ranking function. We finally discuss open issues regarding the use of the unlabeled data during training a ranking function.

1 Introduction

Many learning applications are concerned with the ranking of objects from a fixed collection \mathcal{C} upon a given query. This is the case, for example, in document retrieval or metasearch where the goal is to rank documents from a collection (i.e. the Web or intranets) based on their relevancy to a user's query. Another example is the automatic text summarization task seen as the extraction of relevant text-spans (sentences or paragraphs), where the user provides a document and the system returns a ranked list of text-spans from that document where the top ranked spans are supposed to reflect most, the content of the document. Although there is an increasing interest for the application of Machine Learning algorithms in these domains, the proposed learning settings lack a clear statistical framework.

In the other hand, most computational models proposed for ranking rely only on labeled training examples and ignore the possible information brought from unlabeled data. While a reasonable accuracy may be reached after training such a system on a large set of labeled data, labeling large amounts of data for learning may require expensive human resources and is often unrealistic. In this paper we propose a general setting for learning scoring functions for ranking which handles the additional information provided by unlabeled data for learning. The aim is to show that unlabeled data can help the learning process. Up to our knowledge this is the first time that the use of unlabeled data for ranking has been considered.

This paper is organized as follows; we first show how the Information Retrieval applications we consider could be handled by the proposed approach (section 2) and then present in section 3, the extension of the cross-validation bound of [4] to the ranking.

2 Ranking in Information Retrieval tasks

We consider Information Retrieval (IR) tasks where the system gets a user query $x \in \mathcal{X}$ and must return a subset \mathcal{C}_x of a given collection \mathcal{C} , ordered in such a way that the first elements presented to the user should be the more relevant to his/her query. We consider the problem of learning a function that ranks the elements of \mathcal{C}_x given x . If we furthermore assume **(1)** that there exist a fixed indexing of all subsets \mathcal{C}_x of the form $\mathcal{C}_x = \{z^0, \dots, z^{N_x-1}\}$, where N_x is the number of elements in \mathcal{C}_x , and **(2)** that for all x , $N_x \leq N$, the problem can be expressed as learning a function $\bar{f} : \mathcal{X} \mapsto \sigma_N$, where σ_N is the set of all permutations of $\{0, \dots, N-1\}$, and the ranked list of the elements of \mathcal{C}_x is given by $[z^{\bar{f}(x)^{-1}(i)}]_{i=0..N_x-1}$, where $\bar{f}(x)^{-1}$ is the inverse of $\bar{f}(x)$ ¹.

When learning \bar{f} in a supervised setting, the training set \mathcal{S} generally takes the form of n queries x_i , together with vectors $y_i = (y_i^k)$, $y_i^k \in \{-1, 1\}$, $0 \leq k \leq N_{x_i} - 1$, where y_i^k corresponds to a binary relevance judgement of $z_i^k \in \mathcal{C}_{x_i}$. Although this may not include all IR tasks, we restrict ourselves to this kind of supervision in the paper, since it greatly simplifies the presentation and encompasses most practical cases. Denoting $\mathcal{Y} = \bigcup_{k \leq N} \{-1, 1\}^k$, and $\Delta : \mathcal{Y} \times \sigma_N \mapsto \mathcal{R}_+$ being a risk function, we can define in a standard way the quantities describing the learning task. Given a fixed but unknown distribution \mathcal{D} on $\mathcal{X} \times \mathcal{Y}$, the true (or generalization) risk of \bar{f} is $\epsilon(\bar{f}) = \mathbb{E}_{(x,y) \sim \mathcal{D}} \Delta(y, \bar{f}(x))$, which can be estimated on $\mathcal{S} = (x_i, y_i)_{i=1}^n$ (assuming the (x_i, y_i) s are drawn i.i.d. according to \mathcal{D}) by the empirical risk $\epsilon(\bar{f}, \mathcal{S}) = \frac{1}{n} \sum_{i=1}^n \Delta(y_i, \bar{f}(x_i))$.

We will be interested in estimating $\epsilon(\bar{f})$, where \bar{f} is chosen based on its training set \mathcal{S} , using a set of *unlabeled data* $\mathcal{S}_u = (x'_j)_{j=1}^m$, assumed to be drawn i.i.d. according to $\mathcal{D}_{\mathcal{X}}$, the marginal of \mathcal{D} on \mathcal{X} (notice that together with the x'_j s, the learner has access to $\mathcal{C}_{x'_j}$ even for the unlabeled example). As we will see, such bounds are highly related to the loss function used, which we fix to the following for its convenience in the analysis and in practical applications $\Delta(y, f(x)) = \frac{1}{p_x q_x} \sum_{i: y^i=1} \sum_{j: y^j=-1} [[f(x)(i) > f(x)(j)]]$ where p_x is the number of elements z^k of \mathcal{C}_x with $y^k = 1$ and $q_x = N_x - p_x$, and $[[pr]]$ is one if predicate pr holds and zero otherwise.

Practical issues In practice, a convenient way to learn the function \bar{f} is to learn a scoring function $f : \mathcal{Z} \subset \mathbb{R}^d \mapsto \mathbb{R}$, using a mapping $\Phi : \mathcal{X} \times \mathcal{C} \mapsto \mathcal{Z}$, which is a joint representation of x and an element of \mathcal{C}_x . Given an input x , $\bar{f}(x)$ can be induced from f such that its restriction to $\{1, \dots, N_x - 1\}$ is the permutation of $\{0, \dots, N_x - 1\}$ which satisfies $\bar{f}(x)(i) < \bar{f}(x)(j) \Leftrightarrow f(\Phi(x, z^i)) > f(\Phi(x, z^j))$ (ties are broken arbitrarily), and then setting $\bar{f}(x)(i) = i$ for $i \geq N_x$. For example, in metasearch, \mathcal{C} is the document collection, x a query, \mathcal{C}_x a predefined set of documents extracted from the various search engines, and, for $z \in \mathcal{C}_x$, $\Phi(x, z)$ can be a vector where each dimension is the score returned by a given search engine, and we learn a scoring function f which is a combination of the scores. In extractive text summarization, \mathcal{C} would be the set of all sentences appearing in a document collection, x a document, \mathcal{C}_x the set of sentences appearing in document x , and $\Phi(x, z)$ is a vector of various heuristic scores, each one measuring how much the sentence z satisfies some criterion indicative of whether z reflects the content of x or not.

In terms of optimization, one can learn for example a discriminative linear function of the form $f(\Phi(x, z)) = w \cdot \Phi(x, z)$ with $w \in \mathbb{R}^d$. If we consider, for a given x , two elements z^i and z^j of \mathcal{C}_x such that $y^i = 1$ and $y^j = -1$, we have that $[[\bar{f}(x)(i) > \bar{f}(x)(j)]] = [[w \cdot (\Phi(x, z^j) - \Phi(x, z^i)) > 0]]$, and the empirical risk with the loss Δ resembles a pairwise classification loss of all such pairs z^i, z^j on the training set. Standard classification algorithms can then be adapted to minimize the empirical risk defined in the previous sec-

¹When $N_x < N$, we restrict $\bar{f}(x)(i) = i$ for $N_x \leq i \leq N - 1$.

tion. An example of such an optimization for the case of extractive text summarization can be found in [1]. By the same argument, even an algorithm like RankBoost [2] has to be adapted to be used in information retrieval in order to take into account this difference with the standard pairwise classification setting for ranking.

3 cross-validation bounds for ranking in IR tasks

A possible use of unlabeled data for classification has been introduced by [4]. The main idea of his work is to define a notion of distance between two classifiers c_1 and c_2 that can be computed only based on the unlabeled examples. Basically, if we consider a classification task where examples are in a space \mathcal{X} , have their labels in some space \mathcal{Y} , and if we have a classification loss function $\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}_+$ such that ℓ verifies the triangle inequality (e.g. the 0–1 loss), we have that $\forall (x, y) \in \mathcal{X} \times \mathcal{Y}, \ell(c_1(x), y) \leq \ell(c_2(x), y) + \ell(c_1(x), c_2(x))$. Under the standard assumption that the data (x, y) are drawn according to some probability distribution \mathcal{D} over $\mathcal{X} \times \mathcal{Y}$ and denoting $\mathcal{D}_{\mathcal{X}}$ its marginal on \mathcal{X} , the quantity $\mathbb{E}_{x \sim \mathcal{D}_{\mathcal{X}}} \ell(c_1(x), c_2(x))$ bounds the difference of generalization error between c_1 and c_2 and can be estimated using a set of unlabeled examples.

If we want to borrow the same idea in our case of ranking, we have to face the fundamental difference between our framework and classification, which is that the output of the function we learn is not in the same space as the labels associated to the examples. The function Δ defines a loss between a partial ranking (relevant elements must be presented before irrelevant ones), while the function outputs a total ranking over the elements. As a consequence, there is even no way of defining a triangle inequality. However, the core of the idea of [4] is kept if we can define a quantity $D(\bar{f}, \bar{f}')$, which can be estimated on an unlabeled sample given two ranking functions \bar{f} and \bar{f}' , and which satisfies $\epsilon(\bar{f}) \leq \epsilon(\bar{f}') + D(\bar{f}, \bar{f}')$. The next subsection aims at defining such a D , while the following one follows the same steps as [Matti] to show a cross-validation bound for our cases of ranking.

definition of D We consider here a fixed example x , together with its label $y = (y^1, \dots, y^K)$ where $K = N_x$, and two fixed ranking functions \bar{f} and \bar{f}' . We will denote by p the number of y^i s with value 1 and q the number of them of value -1 . We have then $\Delta(y, \bar{f}(x)) = \frac{1}{pq} \sum_{i:y^i=1} \sum_{j:y^j=-1} [[\bar{f}(x)(i) > \bar{f}(x)(j)]]$. We will furthermore use the following additional notations: $rg(i) = \bar{f}(x)(i)$, which represents the rank (starting from 0) of the element z^i of \mathcal{C}_x in the ranked list output by \bar{f} , and, for i such that $y^i = 1$, $rg_+(i) = \sum_{k:y^k=1} [[rg(k) < rg(i)]]$, which counts the number of relevant elements ranked before the relevant element z^i . We define in the same way the quantities $rg'(i)$ and $rg'_+(i)$ for \bar{f}' .

Since our definitions of ranks start at zero, we have that $\Delta(y, \bar{f}(x)) = \frac{1}{pq} \sum_{i:y^i=1} (rg(i) - rg_+(i))$. Furthermore, we can notice that $\sum_{i:y^i=1} rg_+(i)$ only depends on p (and not on \bar{f} , since it represents the relative rankings of the relevant elements only. As a consequence, we can notice that $\Delta(y, \bar{f}(x)) - \Delta(y, \bar{f}'(x)) = \frac{1}{pq} \sum_{i:y^i=1} (rg(i) - rg'(i))$. Denoting by $\delta(\bar{f}(x), \bar{f}'(x))$ the list of length K containing all the values of $rg(i) - rg'(i)$ for $1 \leq i \leq K$ ordered in decreasing value, and letting $\delta(\bar{f}(x), \bar{f}'(x))_k$ the k -th element of the list (i.e. the k -th greatest value of $rg(i) - rg'(i)$ for $i \in \{1, \dots, K\}$), it is easy to check that:

$$\Delta(y, \bar{f}(x)) - \Delta(y, \bar{f}'(x)) \leq \max_{p,q:p+q=K} \frac{1}{pq} \sum_{k=1}^p \delta(\bar{f}(x), \bar{f}'(x))_k \quad (1)$$

Finally denoting by $d(\bar{f}(x), \bar{f}'(x)) = \max_{p,q:p+q=N_x} \frac{1}{pq} \sum_{k=1}^p \delta(\bar{f}(x), \bar{f}'(x))_k$ and $D(\bar{f}, \bar{f}') = \mathbb{E}_{x \sim \mathcal{D}_{\mathcal{X}}} d(\bar{f}(x), \bar{f}'(x))$, we have that $D(\bar{f}, \bar{f}')$ can be estimated using a set

of unlabeled examples, and, taking the expectation over (x, y) drawn according to \mathcal{D} of equation (1), we have $\epsilon(\bar{f}) \leq \epsilon(\bar{f}') + D(\bar{f}, \bar{f}')$.

A generalization error bound based on cross-validation for ranking² Like in classification, we define a randomized ranking function as being a σ_N -valued random variable that may depend on the input x but is independent from other randomized ranking function. In our work, we will only consider randomized ranking functions \bar{f}_Θ defined by a finite set of ranking functions $\{\bar{f}_k : \mathcal{X} \mapsto \sigma_N, k = 1, \dots, K\}$ and a $\{1, \dots, K\}$ -valued random variable Θ , independent of all other random variables considered. When given an input instance x , a randomized ranking function choses a value $\theta \in \{1, \dots, K\}$ according to the distribution Θ , and outputs $\bar{f}_\theta(x)$. When given a set of instances $x_i, i = 1, \dots, n$, a value θ_i is drawn for each x_i , and a new copy of Θ is used each time, such that all the θ_i are independent of each other. Of course, a deterministic ranking function \bar{f} as considered in the previous sections in the paper is a particular randomized ranking function, where the set contains only one element $\bar{f}_1 = \bar{f}$ and the random variable $\Theta = 1$ with probability 1. The notion of true risk extends to randomized ranking functions by setting $\epsilon(\bar{f}_\Theta) = \mathbb{E}_{\theta \sim \Theta} \mathbb{E}_{(x,y) \sim \mathcal{D}} \Delta(y, \bar{f}_\theta)$. Given two randomized ranking functions \bar{f}_Θ and \bar{f}'_Λ , the function D defined previously also extends, by setting $D(\bar{f}_\Theta, \bar{f}'_\Lambda) = \mathbb{E}_{\theta \sim \Theta, \lambda \sim \Lambda} \mathbb{E}_{x \sim \mathcal{D}_\mathcal{X}} d(\bar{f}_\theta(x), \bar{f}'_\lambda(x))$. Notice that given a realization θ of Θ and λ of Λ , we have $\Delta(y, \bar{f}_\theta(x)) - \Delta(y, \bar{f}'_\lambda(x)) \leq d(\bar{f}_\theta(x), \bar{f}'_\lambda(x))$, and therefore $\epsilon(\bar{f}_\Theta) \leq \epsilon(\bar{f}'_\Lambda) + D(\bar{f}_\Theta, \bar{f}'_\Lambda)$.

Now consider a deterministic ranking function \bar{f} that we just learned, and suppose there is a randomized ranking function \bar{f}'_Θ , for which we can compute a tight bound $\hat{\epsilon}(\bar{f}'_\Theta, \delta/2)$ on $\epsilon(\bar{f}'_\Theta)$ which holds with probability at least $1 - \delta/2$ on the quantities needed to obtain the bound (e.g. the choices of the test set) and such that $D(\bar{f}, \bar{f}'_\Theta)$ may be small. The main idea is that if we can estimate an upper bound on $D(\bar{f}, \bar{f}'_\Theta)$, we will have a tight bound for \bar{f} . Then, assume we have a set $S_u = (x'_j)_{j=1}^m$ of unlabeled examples drawn i.i.d. according to $\mathcal{D}_\mathcal{X}$. Then, if we consider a vector $\theta = (\theta_1, \dots, \theta_m)$ drawn according to Θ^m , $\frac{1}{m} \sum_{j=1}^m d(\bar{f}(x), \bar{f}'_{\theta_j}(x))$ is an unbiased estimate of $D(\bar{f}, \bar{f}'_\Theta)$ that can easily be computed. One can then use a large deviation inequality, like McDiarmid's theorem for bounded differences [5] (since $d(\bar{f}(x), \bar{f}'_{\theta_j}(x)) \in [0, 1]$), to obtain a computable upper bound $\hat{D}(\bar{f}, \bar{f}'_\Theta, S_u, \delta/2)$ on $D(\bar{f}, \bar{f}'_\Theta)$ which holds with probability $1 - \delta/2$ over the possibles S_u and $\theta = (\theta_1, \dots, \theta_m)$. Then, with probability at least $1 - \delta$, we have:

$$\epsilon(\bar{f}) \leq \hat{\epsilon}(\bar{f}'_\Theta, \delta/2) + \hat{D}(\bar{f}, \bar{f}'_\Theta, S_u, \delta/2) \quad (2)$$

Finalizing the adaptation of [4] to our case, an interesting candidate for \bar{f}'_Θ is the randomized ranking function obtained by cross-validation \bar{f}^{cv}_Θ defined as follows. Given a labeled training set $S_l = (x_i, y_i)_{i=1}^n$, used to learn \bar{f} with some algorithm, we arbitrarily split it into K disjoint parts $S_l^k, k = 1, \dots, K$, each of size $n_k = \lfloor \frac{1}{K} \rfloor$ or $n_k = \lfloor \frac{1}{K} \rfloor + 1$ (such that $\sum_k n_k = n$), and, using the same algorithm as for \bar{f} , we train the ranking functions \bar{f}_k on $\bigcup_{k' \neq k} S_l^{k'}$. \bar{f}^{cv}_Θ is then the randomized ranking function having $\{\bar{f}_k, k = 1, \dots, K\}$ as set of ranking functions, and $\Theta = k$ with probability $\frac{1}{K}$ for $k = 1, \dots, K$. The main interest of \bar{f}^{cv}_Θ is that since each \bar{f}_k is trained using the same algorithm and a substantial part of the examples used to train \bar{f} , it is expected that on a given unlabeled set S_u , $\hat{D}(\bar{f}, \bar{f}^{cv}_\Theta, S_u, \delta/2)$ will be small. Furthermore, given $k \in \{1, \dots, K\}$, \bar{f}_k is independent of S_l^k , which can be used as a test set. McDiarmid's theorem can once again be used to obtain an upper bound $\hat{\epsilon}(\bar{f}_k, S_l^k, \frac{\delta}{2K})$ on $\epsilon(\bar{f}_k)$ based on its empirical estimate $\frac{1}{n_k} \sum_{x_i, y_i \in S_l^k} \Delta(y_i, \bar{f}_k(x_i))$, which holds with probability at least $1 - \frac{\delta}{2K}$ over the S_l^k . It is now easy to verify that

²This paragraph contains the straightforward extensions to the work of [4] needed to derive the bound.

$\hat{\epsilon}(\bar{f}_{\Theta}^{cv}, \delta/2) = \frac{1}{K} \sum_{k=1}^K \hat{\epsilon}(\bar{f}_k, S_l^k, \frac{\delta}{2K})$ is an upper bound on $\epsilon(\bar{f}_{\Theta}^{cv})$ which holds with probability $1 - \delta/2$ over the training sets S_l . Replacing \bar{f}'_{Θ} by \bar{f}_{Θ}^{cv} in equation (2) gives the desired computable generalization bound for f learned on S_l , using a set of unlabeled examples S_u .

4 Conclusion and Discussion

In this paper we showed some encouraging results by extending Kaariainen's work [4] to derive generalization error bounds based on cross-validation for a ranking function. We showed in particular that unlabeled data can be used with statistical guarantees for ranking in IR. However, the work in [4], going far beyond this specific bound, is mainly based on an embedding of the classifiers that can be learned and the target $\mathbb{P}(Y|X)$ in a pseudo-metric space of randomized classifiers. But the natural definition of randomized ranking functions is less convenient than randomized classifiers, since in the case of ranking, we do not have an equivalent for the target $\mathbb{P}(Y|X)$ that can be thought of as a randomized ranking function. The same kind of embedding becomes impossible, which shows that such natural extensions of classification results to ranking can not be done in a general way.

The framework we proposed in section 2, close to the label ranking framework proposed by [3], is largely sufficient to describe the supervised task of ranking in IR. However, it seems that specific analysis should be made for different loss functions in a semi-supervised setting. Due to the huge potential of unlabeled data for ranking in practical IR applications, we believe that it is a major issue to define a novel framework to deal with learning problems where the labeling of the examples is not in the domain of values of the function that we learn in which the semi-supervised ranking could be analyzed in a general way. We leave this as a direction for future work.

Acknowledgments

This work was supported by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors views.

References

- [1] Amini M.R., Usunier N., Gallinari P., (2005) Automatic Text Summarization Based on Word Clusters and Ranking Algorithms, *Proceedings of the 27th European Conference on Information Retrieval*.
- [2] Dekel O., Manning C., and Singer Y., (2003) Log-Linear Models for Label Ranking. *NIP 2003*.
- [3] Freund Y., Iyer R.D., Schapire R.E., Singer Y. (2003) An Efficient Boosting Algorithm for Combining Preferences, *Journal of Machine Learning Research 4*, pp. 933-969.
- [4] Kääriäinen M. (2005) Generalization Error Bounds Using Unlabeled Data. *COLT'05*, pp. 127–142.
- [5] McDiarmid C. (1989) On the method of bounded differences, *Surveys in Combinatorics*.