

# A Selective Sampling Strategy for Label Ranking

Massih Amini<sup>1</sup>, Nicolas Usunier<sup>1</sup>, François Laviolette<sup>2</sup>, Alexandre Lacasse<sup>2</sup>,  
and Patrick Gallinari<sup>1</sup>

<sup>1</sup> Laboratoire d'Informatique de Paris 6  
Université Pierre et Marie Curie, Paris, France  
{FirstName.LastName}@lip6.fr

<sup>2</sup> Département IFT-GLO  
Université Laval, Sainte-Foy (QC), Canada  
{FirstName.LastName}@ift.ulaval.ca

**Abstract.** We propose a novel active learning strategy based on the compression framework of [9] for label ranking functions which, given an input instance, predict a total order over a predefined set of alternatives. Our approach is theoretically motivated by an extension to ranking and active learning of Kääriäinen's generalization bounds using unlabeled data [7], initially developed in the context of classification. The bounds we obtain suggest a selective sampling strategy provided that a sufficiently, yet reasonably large initial labeled dataset is provided. Experiments on Information Retrieval corpora from automatic text summarization and question/answering show that the proposed approach allows to substantially reduce the labeling effort in comparison to random and heuristic-based sampling strategies.

## 1 Introduction

This paper presents an active learning strategy for label ranking functions - mappings from instances to rankings over a finite set  $\mathcal{A}$  of *alternatives*. The supervised learning of label-ranking functions has attracted considerable attention from the Machine Learning (ML) community (see e.g. [3,5]) since it encompasses tasks ranging from multi-class(-label) classification to ranking for Information Retrieval (IR) applications.

In this study we are interested on IR-like applications, such as Document Retrieval (DR). In this case, an instance  $x$  is a query and the label  $\ell$  of  $x$  is a partial order over a given document collection  $\mathcal{A}$ . In a supervised setting, the aim is to learn a mapping (or a ranking function) from a predefined set of queries for which there exists a set of *relevance judgments* that indicates which documents in  $\mathcal{A}$  are relevant to each query. In such a case labeling an instance often requires an expert to carefully examine the set of alternatives. The human effort to create labeled datasets may in general be unrealistic. It is thus necessary to design accurate methods for reducing the size of the required labeled set.

Different strategies have been proposed in the classification framework to cope with this kind of problem. One approach is selective sampling: given an

input pool or stream of unlabeled examples, the algorithm selects a few of them and queries an oracle to obtain their labels. These new labeled examples are then added to the training set. Such strategies have been developed around two main ideas. (a) The shrinking of the version space, which in the case of linear discriminant functions, consists in the selection of the unlabeled instance with the smallest margin [11], and (b) the selection of examples which will reduce an approximation of the generalization error [4].

These theoretical motivations, unfortunately, do not extend to label ranking functions. Indeed, there is no equivalent notion of the version space, and approximations of the generalization error are mostly unknown. For the specific case where all the labels of instances are total orders and when the ranking is predicted by a real-valued scoring function, the notion of margin may be extended. Hence, [2] showed that by taking the minimum difference of scores between two alternatives, and selecting the unlabeled examples with the smallest extended margin is a performing heuristic. However, although the "extended margin" heuristic can be also applied in the general case, we cannot expect it to perform well: taking the example of DR, a real-valued scoring function may assign very similar scores to two relevant or two irrelevant documents. Hence, for a given instance, the extended margin may be close to zero independently from the fact that relevant documents have higher scores than irrelevant ones.

In this paper, we propose a new selective sampling strategy for label ranking. Our starting point is the generalization error bounds using unlabeled data proposed by [7] for classification: the generalization error of a classifier can be bounded by the error of another classifier plus the probability of disagreement between both classifiers. In the specific framework of label ranking described in section 2, we show in section 3 that the bounds proposed by [7] can be extended for label ranking in the following way: given a fixed, but arbitrary, cost function and given some prior knowledge about the labels, a cost-specific notion of disagreement between two ranking functions can be constructed. Then the generalization error of a ranking function  $\bar{f}$  we want to learn can be bounded by two terms: the generalization error of a specific ranking function  $\bar{f}^{cv}$  built using cross-validation (CV) sets, and the probability of disagreement between this ranking function and  $\bar{f}$ . We then consider the problem of selective sampling as choosing the unlabeled examples to reduce the generalization error bound of  $\bar{f}$ , which can be done in a greedy fashion by selecting instances for which the disagreement between  $\bar{f}$  and  $\bar{f}^{cv}$  is the highest. Finally, in section 4, we show experimental results on two IR corpora from automatic text summarization and question/answering systems comparing our approach to the extended margin heuristic and the random sampling strategy.

## 2 Notation

Let us define the following notations in addition to those given in the introduction. For simplicity, we identify the set of alternatives  $\mathcal{A}$  by  $\{1, \dots, A\}$ . The instance and the label spaces are denoted respectively by  $\mathcal{X}$  and  $\mathcal{L}$ . A ranking

function is defined as  $\bar{f} : \mathcal{X} \rightarrow \sigma_A$ , where  $\sigma_A$  is the set of permutations of  $\{1, \dots, A\}$ . Hence for an instance  $x \in \mathcal{X}$ , an alternative  $i \in \mathcal{A}$  is preferred over an alternative  $j \in \mathcal{A}$  iff  $\bar{f}(x)(i) < \bar{f}(x)(j)$ . We furthermore suppose that the training set is composed of a *labeled* set  $Z_\ell = ((x_i, \ell_i))_{i=1}^n \in \mathcal{Z}^n$  and an *unlabeled* set  $X_U = (x'_j)_{j=n+1}^{n+m} \in \mathcal{X}^m$ , where  $\mathcal{Z}$  represents the set of  $\mathcal{X} \times \mathcal{L}$ . We suppose that each pair  $(x, l) \in Z_\ell$  is drawn i.i.d with respect to a fixed but unknown distribution  $\mathcal{D}$  and we denote the marginal distribution over  $\mathcal{X}$  by  $D_{\mathcal{X}}$ .

We will furthermore assume that for each instance  $x$ , only a subset  $\mathcal{A}_x$  of  $\mathcal{A}$  is considered, and that  $\mathcal{A}_x$  is known even if the label of  $x$  is unknown. The set of possible labels for  $x$ , denoted  $\mathcal{L}_x$ , contains thus only preference relations over  $\mathcal{A}_x$ . When the labels are induced from binary relevance judgements, any label of  $\mathcal{L}_x$  can be represented by two sets of indices  $Y_x^+$  and  $Y_x^-$  of relevant and irrelevant alternatives in  $\mathcal{A}_x$ .

These notations allow us to formulate naturally costs functions in IR. For example, precision at  $k$  which counts the proportion of relevant alternatives in the first  $k$  positions can be defined by:

$$c_{p@k}(\bar{f}(x), \ell) = \frac{1}{k} \sum_{i \in Y_x^+} \mathbb{I}[\bar{f}(x)(i) \leq k] \quad (1)$$

where  $\mathbb{I}[pr]$  is one if predicate  $pr$  holds and 0 otherwise. Another example is the rank loss function which measures the mean number of irrelevant elements better ranked (the lower the better) by  $\bar{f}$  than relevant ones:

$$c_{Rloss}(\bar{f}(x), \ell) = \frac{1}{|Y_x^+| |Y_x^-|} \sum_{j \in Y_x^-} \sum_{i \in Y_x^+} \mathbb{I}[\bar{f}(x)(j) < \bar{f}(x)(i)] \quad (2)$$

Finally we will denote by  $\hat{e}_Z(\bar{f}) = \frac{1}{n} \sum_{i=1}^n c(\bar{f}(x_i), \ell_i)$  the empirical risk of a ranking function  $\bar{f}$  and by  $\epsilon(\bar{f}) = \mathbb{E}_{(x, \ell) \sim \mathcal{D}} c(\bar{f}(x), \ell)$  its true risk. When  $\bar{f}$  predicts outputs based on a real-valued (i.e. scoring) function over the set  $\mathcal{A}_x$ , we denote by  $f$  the associated scoring function.

### 3 A New Query Selection Strategy

In this section, we present a divergence measure for ranking functions, and present our ranking bounds based on unlabeled data. From that we propose the ranking active learning algorithm which is the central point of the paper.

#### 3.1 Generalization Error Bound for Ranking Functions

We define a randomized ranking function as a  $\sigma_A$ -valued random variable such that for each instance  $x$ , a randomized ranking function  $\bar{f}_\theta$  is chosen according to a probability distribution  $\Theta$  over a finite set of ranking functions  $\{\bar{f}_1, \dots, \bar{f}_K\}$  and an ordered list  $\bar{f}_\theta(x)$  over  $\mathcal{A}$  is returned. If  $\Theta$  is a uniform distribution we denote the randomized ranking function by  $\bar{f}_K$ . The generalization error bound we propose in the following is based on a divergence function  $d_c : \sigma_A \times \sigma_A \rightarrow [0, 1]$

associated with a risk function  $c$  measuring for any query  $x \in \mathcal{X}$  the disagreement between two ranking functions  $\bar{f}$  and  $\bar{f}'$ . We define  $d_c$  as

$$d_c(\bar{f}(x), \bar{f}'(x)) = \max_{\ell \in \mathcal{L}_x} [c(\bar{f}(x), \ell) - c(\bar{f}'(x), \ell)]$$

Clearly,  $d_c$  is a divergence upper bounded by 1 (i.e.,  $1 \geq d_c(y, y') \geq 0$  for any  $y, y'$  and  $d_c(y, y') = 0$  iff  $y = y'$ ). Moreover, we have:

$$\forall (x, \ell) \in \mathcal{Z}, c(\bar{f}(x), \ell) \leq c(\bar{f}'(x), \ell) + d_c(\bar{f}(x), \bar{f}'(x))$$

For two randomized ranking functions  $\bar{f}_\Theta$  and  $\bar{f}'_\Lambda$  the notion of risk can be extended by denoting  $c(\bar{f}_\Theta(x), \ell) = \mathbb{E}_{\theta \sim \Theta} c(\bar{f}_\theta(x), \ell)$  and  $d_c(\bar{f}_\Lambda(x), \bar{f}'_\Theta(x)) = \mathbb{E}_{\lambda \sim \Lambda, \theta \sim \Theta} d_c(\bar{f}_\lambda(x), \bar{f}'_\theta(x))$ .

From these definitions we still have:

$$\forall (x, \ell) \in \mathcal{Z}, c(\bar{f}_\Lambda(x), \ell) \leq c(\bar{f}'_\Theta(x), \ell) + d_c(\bar{f}_\Lambda(x), \bar{f}'_\Theta(x)) \quad (3)$$

Equation 3 shows a link between the values of the risk function  $c$  on  $\bar{f}_\Lambda$  and on  $\bar{f}'_\Theta$  and therefore indicates a possible link between the risk of those two (possibly random) ranking functions. In the stochastic case, the true and empirical risks of a randomized ranking function  $\bar{f}_\Theta$  are defined as

$$\hat{\epsilon}_Z(\bar{f}_\Theta) = \frac{1}{n} \sum_{i=1}^n c(\bar{f}_\Theta(x_i), \ell_i) = \mathbb{E}_{\theta \sim \Theta} \frac{1}{n} \sum_{i=1}^n c(\bar{f}_\theta(x_i), \ell_i)$$

$$\epsilon(\bar{f}_\Theta) = \mathbb{E}_{(x, \ell) \sim \mathcal{D}} c(\bar{f}_\Theta(x), \ell) = \mathbb{E}_{\theta \sim \Theta} \mathbb{E}_{(x, \ell) \sim \mathcal{D}} c(\bar{f}_\theta(x), \ell)$$

Notice that if  $Z$  is drawn i.i.d. according to  $\mathcal{D}$ , then  $\hat{\epsilon}_Z(\bar{f}_\Theta)$  is an unbiased estimator of  $\epsilon(\bar{f}_\Theta)$ . We can also notice that, if  $X_{\mathcal{U}}$  is drawn i.i.d. according to  $\mathcal{D}_{\mathcal{X}}$ , the mean of  $d_c(\bar{f}_\Lambda(x'), \bar{f}'_\Theta(x'))$  for  $x' \in X_{\mathcal{U}}$  is an unbiased estimator of  $\mathbb{E}_{x' \sim \mathcal{D}_{\mathcal{X}}} d_c(\bar{f}_\Lambda(x'), \bar{f}'_\Theta(x'))$ . The following theorem is an extension to ranking functions of a classifier risk bound based on unlabeled data introduced in [7].

**Theorem 1.** *For any two (possibly randomized) ranking functions  $\bar{f}_\Lambda$  and  $\bar{f}'_\Theta$ , we have:*

$$\epsilon(\bar{f}_\Lambda) \leq \epsilon(\bar{f}'_\Theta) + \mathbb{E}_{x' \sim \mathcal{D}_{\mathcal{X}}} d_c(\bar{f}_\Lambda(x'), \bar{f}'_\Theta(x'))$$

*Proof:* Using inequality 3, we get the result by taking the expectation over  $(x, \ell) \sim \mathcal{D}$ .

Thus, using unlabeled data, one can obtain an upper bound on the risk of  $\bar{f}_\Lambda$ , if such a bound exists for  $\bar{f}'_\Theta$ . From now, we suppose that one of the ranking function  $\bar{f}'_\Theta$  is obtained by cross-validation on a training labeled data set which is defined as follows.

**Definition 2 (Cross-validation sets).** *Given a labeled dataset  $Z$  drawn i.i.d. according to  $\mathcal{D}$ , a cross-validation (CV) set of size  $K$  is any partition of  $Z$  into  $K$  disjoint subsets  $Z_1, \dots, Z_K$  of equal size<sup>1</sup>. Moreover, for any CV set  $Z_1, \dots, Z_K$ , we associate the sets, for  $i = 1, \dots, K$ ,  $Z_i^{\text{train}} = \bigcup_{j \neq i} Z_j$  and  $Z_i^{\text{test}} = Z_i$ .*

<sup>1</sup> The results contained in this paper remain valid if the CV set size do not divide  $|Z|$ , but to simplify the notation, we have restricted ourselves to the case where it does.

Hence, given a ranking learning algorithm  $\mathcal{R}$ , and a CV set of size  $K$ ,  $\{Z_1, \dots, Z_K\}$  for  $Z$ , a *randomized ranking function obtained by cross-validation* is the randomized function defined by the uniform probability distribution on the set  $\{\mathcal{R}(Z_1^{train}), \dots, \mathcal{R}(Z_K^{train})\}$ . We will denote by  $\bar{f}_K^{cv}$ , the obtained randomized ranking function, and by  $f_j$ , the function  $\mathcal{R}(Z_j^{train})$ . The results of the rest of this section show how the risk bound of such randomized ranking function can be estimated, and then how the bound of Theorem 1 can be computed in practice. Those results are based on the following version of the Hoeffding's bound:

**Theorem 3 (Hoeffding bound).** *Let  $X_1, \dots, X_n$  be  $n$  copies of a  $[0, 1]$ -valued random variable  $X$ , then, for all  $\delta > 0$ :*

$$\mathbb{P}\left(\mathbb{E}X \leq \frac{1}{n} \sum_{i=1}^n X_i + \sqrt{\frac{\ln(1/\delta)}{2n}}\right) > 1 - \delta$$

Combining Theorem 1 and the Hoeffding bound we obtain the following bound for the true risk of  $\bar{f}_K^{cv}$

**Lemma 4.** *Let  $Z$  be drawn i.i.d. according to  $\mathcal{D}$  and let  $\{Z_1, \dots, Z_K\}$  be a CV set of size  $K$  such that  $K$  divides  $n$ . Then, with probability at least  $1 - \delta/2$  over samples  $Z$ , the risk of  $\bar{f}_K^{cv}$  is given by:*

$$\epsilon(\bar{f}_K^{cv}) \leq \frac{1}{K} \sum_{j=1}^K \hat{\epsilon}_{Z_j}(\bar{f}_j^{cv}) + \sqrt{\frac{K}{2n} \ln \frac{2K}{\delta}}$$

*Proof:* Hoeffding bound implies that for all  $j \in \{1, \dots, K\}$  we have

$\mathbb{P}\left(\epsilon(\bar{f}_j^{cv}) > \hat{\epsilon}_{Z_j}(\bar{f}_j^{cv}) + \sqrt{\frac{K}{2n} \ln \frac{2K}{\delta}}\right) \leq \frac{\delta}{2K}$ . The result of the lemma then follows from the union bound theorem:  $\mathbb{P}(\cup A_i) \leq \sum \mathbb{P}(A_i)$ .

The following lemma bounds  $\mathbb{E}d_c(f(x'), \bar{f}_K^{cv}(x'))$  by its expected value computed over a training unlabeled dataset.

**Lemma 5.** *Let  $X_U$  be an unlabeled dataset drawn independently from a labeled dataset  $Z$ , and let  $\bar{f}$  be a ranking function that has been learned independently of a subset  $X_U^{(k)} = \{x'_{j_1}, \dots, x'_{j_k}\}$  of size  $k$  of  $X_U$ . Then we have:*

$$\mathbb{P}\left(\mathbb{E}_{x' \sim \mathcal{D}_X} d_c(\bar{f}(x'), \bar{f}_K^{cv}(x')) \leq \frac{1}{k} \sum_{l=1}^k d_c(\bar{f}(x'_{j_l}), \bar{f}_K^{cv}(x'_{j_l})) + \sqrt{\frac{\ln(\frac{2}{\delta})}{2k}}\right) > 1 - \frac{\delta}{2}$$

where the probability is taken over the choices of  $X_U$ .

*Proof:* Since  $d_c$  is a  $[0, 1]$ -random variable which is function of the  $x' \in X_U^{(k)}$ , and since the  $x'$  are i.i.d., the result follows from Theorem 3, [with  $\delta := \delta/2$ ,  $n := k$ ,  $X := d_c(\bar{f}(x'), \bar{f}_K^{cv}(x'))$ ,  $X_i := d_c(\bar{f}(x'_{j_i}), \bar{f}_K^{cv}(x'_{j_i}))$ ].

Theorem 1, together with the last two lemmas give an upper bound of the risk of any *a priori* chosen ranking function. This bound can be accurately estimated from the labeled data and a subset of the unlabeled data, provided that the size of the latter and  $n/K$  are large enough, and provided that the divergence  $d_c$  can be easily estimated. The next proposition shows that this is the case for the risk function  $c := c_{Rloss}$  that we consider in our experiments.

**Proposition 6.** *Let  $\bar{f}$  and  $\bar{f}'$  be two ranking functions, and  $x$  an unlabeled instance in the case where labels are generated based on binary relevance judgments of the alternatives. Then, using the risk functions of Equation 2, we have:*

$$d_{c_{Rloss}}(\bar{f}(x), \bar{f}'(x)) \leq \max_{p,q:p+q=A_x} \frac{1}{pq} \sum_{k=1}^p \delta(\bar{f}(x), \bar{f}'(x))_k$$

where  $\delta(\bar{f}(x), \bar{f}'(x))$  is the list of length  $A_x$  containing all the values of  $\bar{f}(x)(i) - \bar{f}'(x)(i)$  for  $1 \leq i \leq A_x$  ordered in decreasing value.

*Proof:* Assume that the true label  $\ell$  of  $x$  is  $Y_x = (Y_1, \dots, Y_{A_x})$ . We denote  $rg(i) = \bar{f}(x)(i) - 1$ , and for each  $i \in Y_x^+$ ,  $rg_+(i) = \sum_{j \in Y_x^+} [\bar{f}(x)(i) > \bar{f}(x)(j)]$  (the number of relevant alternatives ranked before relevant alternative  $i$ ). Then, using equation 2, we have  $c_{Rloss}(\bar{f}(x), \ell) = \frac{1}{|Y_x^+||Y_x^-|} \sum_{i \in Y_x^+} (rg(i) - rg_+(i))$ . Since  $rg_+$  and  $rg'_+$  do only consider relevant alternatives, we have  $\sum_{i \in Y_x^+} rg_+(x) = \sum_{i \in Y_x^+} rg'_+(i)$ , and therefore

$$c_{Rloss}(\bar{f}(x), \ell) - c_{Rloss}(\bar{f}'(x), \ell) \leq \frac{1}{|Y_x^+||Y_x^-|} \sum_{i=1}^{|Y_x^+|} \delta(\bar{f}(x), \bar{f}'(x))_i$$

The results yields by taking the maximum value of the right-hand side of the last equation over all possible values of these numbers.

For a given instance  $x$ , the complexity of  $d_{c_{Rloss}}$  is  $O(|\mathcal{A}_x| \ln |\mathcal{A}_x|)$ , since the most expensive computation is sorting a list of size  $|\mathcal{A}_x|$ .

### 3.2 A Uniform Risk Bound for Active Learning

To minimize  $\epsilon(\bar{f}_\Theta)$ , one can try to minimize  $\mathbb{E}_{x' \sim \mathcal{D}_X} d_c(\bar{f}(x'), \bar{f}_K^{cv}(x'))$ . However, in order to use Theorem 1 in an active learning algorithm, we will need a bound that is uniformly valid for all ranking functions  $\bar{f}$ , and all “possible” sequence of queries. This can be done in the same way as for the sample compression scheme [6] in supervised learning via the union bound.

In the sample compression scheme, given a (labeled) training set  $S$  of an *a priori* defined size  $m$ , any classifier returned by a learning algorithm is described by a compression set. A *compression set* is a subset of the training set  $S$  and therefore, when  $S$  is given, can be described as a vector of indices  $\mathbf{i} = (i_1, i_2, \dots, i_k)$  with  $i_j \in \{1, \dots, m\} \forall j$  and  $i_1 < i_2 < \dots < i_k$ . This implies that there exists a deterministic *reconstruction function*, associated with the algorithm, that outputs a classifier when given a training set and a vector of indices. The perceptron and the SVM are such an example. In that setting, given an *a priori* defined vector  $\mathbf{i}$  of indices, one can use the examples of the training set that do not correspond to any index of  $\mathbf{i}$  to bound the risk of the classifier defined by  $\mathbf{i}$  (and the training set  $S$ ). Moreover, provided a prior distribution is given on the set of all possible vector of indices, one can extend such a bound to a bound which is valid simultaneously for all classifiers that can be reconstructed [8].

Since any active learning ranking algorithm  $\mathcal{R}$  considered in this paper is deterministic, the set of all possible ranking functions that can be output by  $\mathcal{R}$  depends only on the set of all examples of  $X_{\mathcal{U}}$  that have been queried during the execution together with all the corresponding labels that have been given in response to the queries. Moreover, if we make the following assumption:

**Assumption 7.** *There exists a deterministic function  $\phi : \mathcal{X} \rightarrow \mathcal{L}$  such that for all  $(x, \ell)$  drawn according to  $\mathcal{D}$ , we have  $\ell = \phi(x)$ .*

The set of all possible ranking functions that can be output by  $\mathcal{R}$  will then depend only on the labeled set  $Z$  together with the final set of all the activated examples. Thus, as for the sample compression scheme, we have a reconstruction function associated with  $\mathcal{R}$ . We can therefore apply the same techniques as for the sample compression scheme to deduce risk bounds that will be valid for all ranker functions that can be reconstructed. The next results formalize this idea.

Starting from the whole set  $X_{\mathcal{U}}$  of unlabeled, minimizing the generalization error of  $\bar{f}$  can then be done by considering a subset of  $X_{\mathcal{U}}^{(k)}$  of  $k$  elements of  $X_{\mathcal{U}}$  for which the value of  $d_c(\bar{f}(x'), \bar{f}_K^{cv}(x'))$  are maximal (Algorithm 1). Then, we can ask for the labels of  $x' \in X_{\mathcal{U}}^{(k)}$  and learn  $\bar{f}$  on  $Z_{\ell} \cup Z_{\mathcal{U}}^{(k)}$ , where  $Z_{\mathcal{U}}^{(k)}$  denotes the labeled dataset, together with examples  $x' \in X_{\mathcal{U}}$  that have been activated.

---

**Algorithm 1.** Active Learning strategy for ranking

---

**Input** :

- A set of labeled  $Z_{\ell}$  and unlabeled examples  $X_{\mathcal{U}}$ ,
- $k$  the number of examples to be activated,  $T$  the maximum number of rounds.

**Initialize:**

- $\forall j \in \{1, \dots, K\}$  learn  $\bar{f}_j^{cv}$  on  $Z_j$ , set  $Z_{\mathcal{U}}^{(k)} \leftarrow \emptyset$  and  $t \leftarrow 1$ .

**repeat**

- Learn  $\bar{f}$  on  $Z_{\ell} \cup Z_{\mathcal{U}}^{(k)}$ ,
- Select a subset  $X_{\mathcal{U}}^{(k)} \subset X_{\mathcal{U}} | \forall x' \in X_{\mathcal{U}}^{(k)}$  the value  $d_c(\bar{f}(x'), \bar{f}_K^{cv}(x'))$  is maximal,
- Ask for the labels of  $x'$  for  $x' \in X_{\mathcal{U}}^{(k)}$ ,
- Remove  $X_{\mathcal{U}}^{(k)}$  from  $X_{\mathcal{U}}$  and reaffected  $Z_{\mathcal{U}}^{(k)}$ ,  $Z_{\mathcal{U}}^{(k)} \leftarrow Z_{\mathcal{U}}^{(k)} \cup X_{\mathcal{U}}^{(k)}$ ,  $t \leftarrow t + 1$

**until** convergence of  $\sum_{x' \in X_{\mathcal{U}}} d_c(\bar{f}(x'), \bar{f}_K^{cv}(x')) \vee t > T$  ;

**Output** :  $\bar{f}$

---

The interested reader may refer to [5] to have descriptions of existing supervised algorithms for learning ranking functions in step 1 of the algorithm.

In the following, we suppose that  $Z = \{(x_1, \ell_1), (x_2, \ell_2), \dots, (x_n, \ell_n)\}$  and  $X_{\mathcal{U}} = \{x'_{n+1}, x'_{n+2}, \dots, x'_{n+m}\}$ . Moreover, we will also suppose that any single query of the ranking active learning algorithm corresponds to an activation of exactly  $k$  unlabeled data (for a parameter  $k$  fixed *a priori*), the total number of

activated examples will then always be of the form  $k \cdot t$  for some  $t \in \mathbb{N}$ . Thus, the compression set of any ranking function related to the algorithm  $\mathcal{R}$  is the union of the set  $Z$  and a subset of size  $k \cdot t$  of  $X_{\mathcal{U}}$ . The set of the labeled data is always in the compression set because the algorithm always consider  $Z$ . Now, one can define a prior distribution on the set of all outputs of  $\mathcal{R}$  by defining a prior  $P_{\mathbb{N}}$  on  $\mathbb{N}$  together with, for each  $t$  that has weight in  $P_{\mathbb{N}}$ , a prior  $P_t$  on the set of all possible vector of indices of the forms  $\mathbf{i} = (1, 2, \dots, n, i_1, i_2, \dots, i_{kt})$  with  $i_j \in \{n + 1, \dots, n + m\} \forall j$  and  $i_1 < i_2 < \dots < i_{kt}$ .

Most of the time, the prior  $P_{\mathbb{N}}$  will have all its weights on the set  $\{1, 2, \dots, T\}$  for some parameter  $T$  defined *a priori*. Moreover, unless  $m$  is too big, since the examples of  $X_{\mathcal{U}}$  are supposed i.i.d., we will choose  $P_t$  as the uniform distribution under the constraint that the  $n$  first indices must always be chosen, that is  $P_t(\mathbf{i}, t) = \binom{m}{kt}^{-1}$  for any  $(\mathbf{i}, t)$ . We will denote by  $\mathcal{R}_{(\mathbf{i}, t)}$  the corresponding ranking function. Under those assumptions, we have the following result.

**Theorem 8.** *Let  $\mathcal{R}$  be any ranking active learning algorithm whose queries are all of size  $k$ . Let  $P_{\mathbb{N}}$  and  $\{P_t\}_{t \in \mathbb{N}}$  be the priors defined above. Finally, let  $\bar{f}_K^{cv}$  be any stochastic ranking function (possibly defined by cross validation on a labeled dataset). Then  $\forall t \in \mathbb{N}$  and  $\forall \mathbf{i} = (1, 2, \dots, n, i_1, i_2, \dots, i_{kt})$  we have:*

$$\mathbb{P} \left( \mathbb{E}_{x' \sim \mathcal{D}_X} d_c(\mathcal{R}_{(\mathbf{i}, t)}(x'), \bar{f}_K^{cv}(x')) \leq \hat{\epsilon}_{Z \cup X_{\mathcal{U}}^{(kt)}} + \sqrt{\frac{\ln \binom{m}{kt} - \ln P_{\mathbb{N}}(t) + \ln(2/\delta)}{2(m-kt)}} \right) > 1 - \frac{\delta}{2}$$

where

$$\hat{\epsilon}_{Z \cup X_{\mathcal{U}}^{(kt)}} \stackrel{\text{def}}{=} \frac{1}{m-kt} \sum_{x' \in X_{\mathcal{U}} \setminus X_{\mathcal{U}}^{(kt)}} d_c(\mathcal{R}_{(\mathbf{i}, t)}(x'), \bar{f}_K^{cv}(x')).$$

*Proof:* Similarly as in the proof of Lemma 4, for each  $(\mathbf{i}, t)$ , we use Hoeffding inequality [with  $\delta := \frac{\delta \cdot P_{\mathbb{N}}(t) \cdot P_t(\mathbf{i}, t)}{2}$ ] and then apply the union bound.

Theorem 8, together with Theorem 1 and Lemma 4 gives us a generalization error bound (with level of confidence  $1 - \delta$ ) on any ranking function learned using this type of active learning procedure. Also it shows that any such  $\mathcal{R}$  will converge to a ranking function that is at least as good as the cv-ranking function  $\bar{f}_K^{cv}$  even if  $\mathcal{R}$  is constructing *deterministic* ranking function only. Moreover, it is clear from the definition of the divergence  $d_c$  that, for any already constructed ranking function  $\bar{f}$ , the corresponding label of any unlabeled data for which the value of  $d_c$  is maximal will gives rise to one of those three situations: (1)–the  $c$ -value of  $\bar{f}_K^{cv}$  is “good” and the one of  $\bar{f}$  is “bad”, (2)–the  $c$ -value of  $\bar{f}_K^{cv}$  is “bad” and the one of  $\bar{f}$  is “good”, or (3)–both are “bad”. Clearly a query in the situation (1) or (3) points out a weakness of  $\bar{f}$ . From an active learning point of view, this is something that is suitable. Note also that, if  $\bar{f}_K^{cv}$  has a low risk, then situations (1) and (3) would be more likely to occur. This is the central idea that underlies our proposed ranking active learning algorithm<sup>2</sup>.

<sup>2</sup> As in [7], we choose to base our approach on the generally accepted assumption that CV methods give good results.

## 4 Experiments

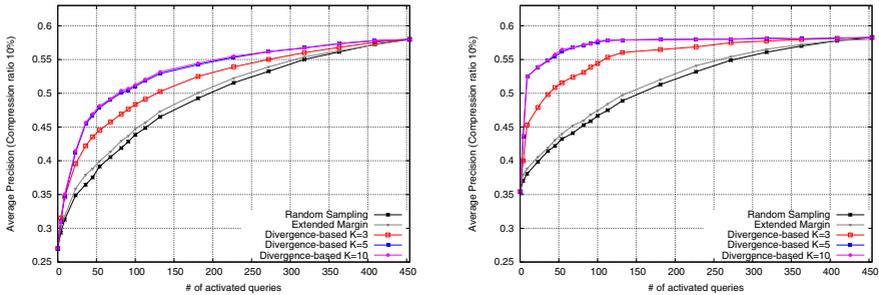
We compared the proposed selective sampling scheme (denoted by *divergence-based* in the following) with the random sampling and the extended margin heuristic of [2] adapted to partial orderings. The reference supervised learning algorithm we used to train the randomized ranking functions is the same as in [1]. For a ranking function  $\bar{f}$ , we used the  $c_{Rloss}$  risk function to learn a linear combination of weights for its associated scoring function  $f$ . We employed the divergence measure  $d_{c_{Rloss}}$  introduced in proposition 6 to activate queries from  $X_U$  and conducted experiments on the Information Retrieval tasks of text-summarization (TS) and question/answering (QA). Performances for TS and QA are respectively averaged over 10 and 25 random splits<sup>3</sup> of training/unlabeled pool/test sets. For the text summarization, the queries we aim to activate are documents for which the list of sentences appearing in its summary is demanded. For QA, queries are questions and for each activated question we ask for passages containing its answer.

### 4.1 Real Life Applications

*Automatic Text Summarization.* Automatic Text Summarization (ATS) systems are mostly designed to help users to quickly find a needed information. Most studies consider the task of text summarization as the extraction of text spans (typically sentences) from the original document. Extractive approaches transform the problem of abstracting a text into a simpler problem of *ranking* sentences according to their relevance of being part of the document summary. These approaches have proven to be effective in producing summaries. To rank text spans from a document, most previous studies combine statistical or linguistic features characterizing each sentence in a text. A combination of these features is finally used to order the spans. In this work we considered 4 statistical features borrowed from [1]. For ATS, we compared performance on the WIPO collection<sup>4</sup> used in [1]. In our experiments, we have chosen 1000 documents at random from this corpus and removed documents having less than 2 words in their title, and those composed of 1 sentence arguing that a sentence is not sufficient to summarize a scientific document. In total we gathered 854 documents and their associated summaries and Train/Unlabeled/Test splits are respectively 60/394/400 and 30/394/400 in each experiment. For the evaluation we followed the state-of-the-art by comparing the extract of the system for each document in the test collection with the desired summary obtained from its abstract by an alignment technique [10]. We used the *average precision* measure by fixing a 10% compression ratio, that is for each document in the collection, we computed the average number of sentences appearing in its summary in a high ranked sentence list of a length equal to 10% of the document’s size.

<sup>3</sup> We conducted the Wilcoxon rank sum tests to decide of the significance of results for Q/A and thus ran more experiments in this case.

<sup>4</sup> <http://www.wipo.int/ibis/datasets/index.html>



**Fig. 1.** Average Precision at 10% compression ratio versus the number of activated queries for random, extended margin and divergence-based strategies. Results are averaged over 10 randomly splits of training /unlabeled pool/test sets. For the same number of documents in the unlabeled pool (394) and test set (400), performance are plotted for 30 (left) and 60 (right) documents in the training set.

*Passages retrieval for Question/Answering.* QA systems address the problem of finding exact answers to natural language (NL) questions. In order to reduce the amount of information, QA systems apply successively two different modules. A *document retrieval* module first identifies spans (documents or paragraphs) that are likely to contain an answer to the asked question. Then an *answer extraction* module extracts the desired answer by performing a deeper NL analysis of the retrieved spans. Here we consider the *document retrieval* module of a QA system. For Q/A, we compared performance on the TREC-11 question/answering track and the *Aquaint* collection<sup>5</sup> by evaluating the  $a@n$  measure which is the proportion of questions in the test set, for which the answer is contained in the first  $n$  retrieved passages. Among the 500 questions in the collection, we discarded 193 having no answer in the top 100 passages retrieved by the search engine. For each question, we followed the methodology developed in [12] to convert the retrieved passages into 117 dimensional score features by applying a conventional search engine which assigns a series of scores to each paragraph in the collection. In this setting, Train/Unlabeled/Test splits are 30/121/156.

## 4.2 Empirical Results

Figure 1, plots the performance of the divergence-based, extended margin and random strategies for the ATS task for different numbers of randomized ranking functions and for different splits of the training set (training sets of size 30 documents in figure 1. left and 60 documents in figure 1. right - the size of unlabeled data and test sets are kept fixed). We see in both cases that divergence-based strategy has a real advantage over random sampling and the extended margin heuristic. The low performance of the extended margin can be explained by the fact that an accurate scoring function should be able to rank relevant

<sup>5</sup> [http://trec.nist.gov/data/qa/t2002\\_qadata.html](http://trec.nist.gov/data/qa/t2002_qadata.html)

**Table 1.**  $a@n$  in % for the divergence-based, random and the extended margin strategies. Results are averaged over 25 randomly splits of training/unlabeled/test sets.

$n$	Strategy	# of activated queries ( $K = 5$ )					$a@n$
		0	4	8	24	60	
5	Divergence-based	35.2	<b>39.7</b>	<b>41.6</b>	<b>44.8</b>	<b>45.5</b>	46.1
	Extended Margin		38.6 $\leftarrow$	40.1 $\rightleftharpoons$	41.7 $\rightleftharpoons$	44.2 $\leftarrow$	
	Random		38.0 $\rightleftharpoons$	39.6 $\rightleftharpoons$	41.3 $\rightleftharpoons$	43.8 $\rightleftharpoons$	
10	Divergence-based	46.1	<b>51.2</b>	<b>52.5</b>	<b>56.4</b>	<b>57.6</b>	57.7
	Extended Margin		49.9 $\leftarrow$	51.7 $\leftarrow$	54.2 $\rightleftharpoons$	56.9 $\leftarrow$	
	Random		49.5 $\leftarrow$	51.2 $\rightleftharpoons$	53.7 $\rightleftharpoons$	56.2 $\rightleftharpoons$	
20	Divergence-based	53.8	<b>57.7</b>	<b>62.8</b>	<b>67.3</b>	<b>68.6</b>	69.2
	Extended Margin		56.8 $\leftarrow$	60.1 $\rightleftharpoons$	64.9 $\rightleftharpoons$	67.3 $\rightleftharpoons$	
	Random		56.2 $\rightleftharpoons$	59.5 $\rightleftharpoons$	64.4 $\rightleftharpoons$	66.9 $\rightleftharpoons$	

sentences above irrelevant ones, but we should not expect this scoring function to be confident about the relative ranks of two relevant (or two irrelevant) sentences.

In the case where the randomized ranking functions (RRF) have sufficiently been trained (figure 1. right) we note that after querying of about 50 instances with 5 or 10 RRF, the final ranking function has approximately the same level of performance as when the ranking function is learnt on all the labeled data, together with all the unlabeled data and their corresponding labels.

The convergence rate of the performance of deterministic ranking functions is however lower with a smaller number of RRF. This might be due to the fact that the split of the training set on different cross-validation sets (on which each RRF is trained) is larger with a higher number of RRF. Thus the divergence-based strategy appears to be most effective if there is a reasonable training size for learning and not a too small number of RRFs.

Table 2, shows our second investigation for the Q/A task. We notice the same effect of the divergence-based strategy compared to random sampling and extended margin heuristic than for ATS. Indeed, with only 30 questions in the training set, the divergence-based strategy outperforms the random and extended margin strategies for different values of  $n$ . We conducted Wilcoxon rank sum tests with a p-value threshold of 0.05 to decide if the results in Table 2 are significant. The symbols  $\leftarrow$  and  $\rightleftharpoons$  indicate the cases where Extended-Margin and Random strategies are significantly worse than the Divergence-based strategy respectively as a one and two-tailed tests.

## 5 Conclusion

We proposed an active learning strategy for learning ranking functions. The theoretical analysis and the definition of the notion of disagreement between two ranking functions lead to a novel active learning strategy that shows good empirical performance on real world applications. To the best of our knowledge, this strategy is the first one that can be applied to general cases of ranking.

Moreover, experimental results show that, in practice, the derived active learning strategy is highly effective.

The major theoretical weakness of our work is that we consider the randomized ranking function built with CV sets as the reference, while it is certainly not a perfect ranker. Indeed, the generalization error bound can never be better than the generalization error on the CV-ranker. On the other hand, our analysis comes with several advantages: (1) we actually tend to minimize a generalization error bound, which is a challenging issue in label ranking. (2) The bound remains valid during the active learning process. (3) Our proposed sample compression approach gives a general framework on which one can base other ranking active learning algorithms. Finally, it appears empirically that the divergence-based strategy suggested by the bound significantly reduces the required number of labeled examples. Therefore, while the proposed strategy is, indeed, mainly heuristic, it needed the bound for both the definition of the notion of disagreement, and for the idea of using another ranking function as reference. Possible improvements of our theory include the study of how the ranking function used as reference could evolve as we query for more labels, which would enable the generalization error bound of the learned function become better than the initial error of the randomized ranking function obtained with CV-sets.

## References

1. M. Amini, N. Usunier, and P. Gallinari. Automatic text summarization based on word clusters and ranking algorithms. In *Proc. of the 27<sup>th</sup> ECIR*, 2005.
2. Klaus Brinker. Active learning of label ranking functions. In *Proc. of 21<sup>st</sup> International Conference on Machine learning*, 2004.
3. Klaus Brinker, Johannes Fürnkranz, and Eyke Hüllermeier. Label ranking by learning pairwise preferences. In *Journal of Machine Learning Research*, 2005.
4. Olivier Chapelle. Active learning for parzen window classifier. In *AI STATS*, 2005.
5. Koby Crammer and Yoram Singer. A family of additive online algorithms for category ranking. *Journal of Machine Learning Research*, 3(6):1025 – 1058, 2003.
6. Sally Floyd and Manfred Warmuth. Sample compression, learnability, and the Vapnik-Chervonenkis dimension. *Machine Learning*, 21(3):269–304, 1995.
7. Matti Kääriäinen. Generalization error bounds using unlabeled data. In *Proceedings of the 18<sup>th</sup> Annual Conference on Learning Theory*, pages 127–142, 2005.
8. François Laviolette, Mario Marchand, and Mohak Shah. Margin-sparsity trade-off for the set covering machine. *Proc. of the 16<sup>th</sup> ECML*, pages 206–217, 2005.
9. N. Littlestone and Manfred Warmuth. Relating data compression and learnability. *Technical Report, University of California*, 1986.
10. Daniel Marcu. The automatic construction of large-scale corpora for summarization research. In *Proceedings of the 22nd ACM SIGIR*, pages 137–144, 1999.
11. Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *J. Mach. Learn. Res.*, 2:45–66, 2002.
12. N. Usunier, M. Amini, and P. Gallinari. Boosting weak ranking functions to enhance passage retrieval for question answering. In *IR4QA-workshop, SIGIR*, 2004.