Multiview Semi-Supervised Learning for Ranking Multilingual Documents

Nicolas Usunier[1], Massih-Reza Amini[2], and Cyril Goutte[2]

[(1)] Université Pierre et Marie Curie, LIP6
F-75252 Paris 5 cedex, France

[(2)] National Research Council Canada, IIT
Gatineau, QC J8X 3X7, Canada

Abstract. We address the problem of learning to rank documents in a multilingual context, when reference ranking information is only partially available. We propose a multiview learning approach to this semi-supervised ranking task, where the translation of a document in a given language is considered as a view of the document. Although both multiview and semi-supervised learning of classifiers have been studied extensively in recent years, their application to the problem of ranking has received much less attention. We describe a semi-supervised multiview ranking algorithm that exploits a global agreement between view-specific ranking functions on a set of unlabeled observations. We show that our proposed algorithm achieves significant improvements over both semi-supervised multiview classification and semi-supervised single-view rankers on a large multilingual collection of Reuters news covering 5 languages. Our experiments also suggest that our approach is most effective when few labeled documents are available and the classes are imbalanced.

Keywords: Learning to Rank, Semi-supervised Learning, Multiview Learning

1 Introduction

We address the problem of ranking multilingual documents. Ranking is an important problem in several applications related to Information Retrieval such as search, or summarization. Although multilingual document collections are very common in many national or supranational contexts, the bulk of document organization techniques is still developed in a monolingual setting, often for English. We aim at developing ranking tools for handling such multilingual collections in ways smarter than using independent monolingual approaches. We also consider situations where only partial supervision is available in the form of reference ranking information.

In order to learn in a multilingual setting, our proposal relies on the framework of *multiview* learning. In a parallel corpus of multilingual documents, we consider each language as a separate *view* of a document. Each document will therefore have as many views as there are languages in the corpus. Earlier work suggests that this framework is an efficient way to learn classifiers in a multilingual setting [3]. We show how multiview learning can be extended to ranking,

and how it can be applied to ranking multilingual documents. More specifically, we are interested in *bipartite* ranking problems such as information routing [23], in which we seek a linear ordering of objects that belong to two relevance judgments, such that relevant examples are ranked higher than irrelevant ones. This task has been extensively studied in the supervised learning setting [9, 11, 16] due to its practical importance. It is also a first step towards more general ranking tasks, where the reference ranking information can take the form of an arbitrary preference relation over the examples [13]. In addition, a common issue with tasks involving large collections of textual documents is that providing extensive human supervision (such as category labels or ranking information) can be prohibitively expensive. Semi-supervised learning techniques have been developed to address this problem. In the framework of multiview learning for classification, these approaches use the labeled data to train several view-specific predictors, and rely on the intuition that these predictors should have similar predictions on the unlabeled set. This additional constraint may reduce the possible choices of predictors, leading to better generalization guarantees [24]. Our approach to semi-supervised multiview ranking (`SmVR`) follows the same intuition. Given score functions $(h_1, ..., h_V)$ independently trained on each view, we define a notion of global agreement between them as the expectation, over random pairs of objects (x, x'), that two score functions $(h_v, h_{v'})$ predict the same relative ordering. We hence describe a learning process in which language or view-specific ranking functions should achieve high ranking performance on the labeled training set, while minimizing a disagreement measure between each other on the unlabeled dataset.

We propose an efficient multilingual ranking algorithm inspired by iterative co-training techniques [7]. Our method exploits randomization and efficient algorithms for supervised bipartite ranking to break the quadratic complexity (with respect to the number of unlabeled objects) inherent to the `SmVR` approach based on the minimization of the disagreement. Experiments carried out on a multilingual text corpus indicate that `SmVR` provides a significant improvement over both single-view semi-supervised ranking and semi-supervised multiview classification, and is more robust to class imbalance than a state-of-the-art semi-supervised multiview classification algorithm. Promising results have also been published on semi supervised ranking in the single-view setting [2, 14, 22], but to the best of our knowledge, none was extended to multiview learning.

In the next section, we briefly review some related state-of-the-art. In Section 3, we present our solution to semi-supervised ranking in a multiview setting, and Section 4 describes the algorithm applied in our experiments. The experimental results are reported in section 5.2, where we show that our method is effective on a large multilingual collection of Reuters documents covering five languages.

2 Related Work

In this section, we review the state-of-the-art on bipartite ranking, multiview learning and semi-supervised learning for classification and ranking.

2.1 Bipartite Ranking

The task of learning to rank was introduced by Cohen et al. [10], motivated by information retrieval applications where the results take the form of an ordered list of objects. The new framework introduced an algorithm with the ability to learn from a new form of supervision, namely preference relations over the examples. The algorithm also optimized some criteria related to the ranking performance of the predictor. While that original ranking algorithm learned a preference relation on the example space, subsequent proposals reduced the task to learning a scoring function [15, 13]. The ranking is then created by sorting the examples by decreasing scores. Bipartite ranking is the special case of ranking where the supervision is a bipartite graph [13]. It corresponds to information routing problems where the query (or topic) is fixed and examples are either relevant or irrelevant to the query [23]. Bipartite ranking can be formulated as the learning of a scoring function by optimizing the area under the `ROC` curve (`AUC`, see Section 3) [1]. While many classification algorithms produce scores and thus can be used in the context of bipartite ranking, Cortes & Mohri [11] analyze the advantage of optimizing the `AUC` instead of the classification accuracy when one searches good ranking performance. Their conclusion is that ranking methods should be superior when the data is imbalanced (a vast majority of the examples belong to the same class) or noisy. The theory underlying bipartite ranking has been extensively studied [9] and efficient algorithms for `AUC` optimization have been designed [13, 16]. From an algorithmic perspective, the extension of supervised learning algorithms from bipartite ranking to the general case is usually straightforward, even though the computational cost might significantly increase. Most works on bipartite ranking were done in the supervised and single view setting. We propose here an extension to semi-supervised multiview learning.

2.2 Multiview Learning

Multiview learning deals with observations that can be described in several representation spaces, such that each representation space may be used to build a predictor. Multilingual documents can naturally be seen as multiview observations: each language in which a document is translated corresponds to a view. The overall goal of multiview learning is to combine predictors over each view (called *view-specific* predictors) in order to improve the overall performance beyond that of predictors trained on each view separately, or on trivial combinations of views. The first successful multiview learning technique was Blum's co-training algorithm [7] which iteratively labels unlabeled examples based on predictors trained in different views. A related approach is co-regularization [24] where the view-specific predictors are constrained to produce similar predictions. Other notable multiview techniques are multiple kernel learning approach (`MKL`, e.g. [4]) , and techniques relying on (kernel) Canonical Correlation Analysis [18] or multiview Fisher Discriminant Analysis [12]. Note that although co-training [7] and co-regularization [24] have different theoretical backgrounds

and motivation, empirical evidence shows that view-specific classifiers trained by iterative co-training algorithm tend to agree on the pool of unlabeled data. The pseudo-labeling method of co-training can thus be seen as an iterative method for increasing the agreement between predictors. This issue will be at the core of our approach (see Section 4). Although multiview learning has been used from its origin on textual data [7], it has only recently been applied to multilingual data [3]. Moreover, the multiview framework has been extensively studied for classification tasks, but its use in bipartite ranking is novel.

2.3 Semi-supervised Classification

Apart from multiview approaches, the field of single view semi-supervised learning has been an active area of research since the late nineties [27]. The overall aim is to design algorithms which are able to extract information from both labeled and unlabeled data to improve performance. While some work on semi-supervised learning deals with ranking tasks, the main focus was classification. Most studies on the semi-supervised paradigm rely on the *cluster assumption*, which states that examples within a given cluster are likely to be of the same class. Algorithms designed for this assumption are generally based on mixture models [21]. Semi-supervised discriminative approaches are mainly based on a similar but slightly different assumption of *low density separation*, which states that high-density regions do not contain the decision boundary [8]. These approaches are mostly iterative algorithms designed to propagate the class labels in the high density regions. Another marginally different assumption is the *manifold assumption*, which holds when high dimensional data lie on a low-dimensional manifold [6]. In such cases, the learning algorithm can avoid the curse of dimensionality which may affect generative models by operating in a low-dimensional space [5]. While both supervised learning for ranking and semi supervised learning for classification have been widely studied in the past, the combination of semi-supervised learning for ranking has just begun to be explored.

2.4 Single view Semi-Supervised Ranking

Both supervised learning and our approach to multiview, semi-supervised learning of ranking functions in the bipartite setting are inspired by algorithms for binary classification. The approaches to single view semi-supervised learning of classifiers, however, cannot be easily adapted to ranking. Indeed, the assumptions used in single view semi-supervised classification are such that the decision boundary is easy to detect on the set of unlabeled data. The task of ranking, however, is not about detecting a decision boundary, but rather a scoring function that induces the best possible complete ordering of the observations. This ranking is given by scoring the observations according to their probability of being relevant [9], an information that is not considered by classification criteria: these algorithms only need the most probable class label for a given observation. Some work has been done on single view semi-supervised bipartite ranking with promising experimental results. In [2], an iterative pseudo-labeling step

uses neighborhood information while optimizing a ranking objective function on labeled (and pseudo-labeled) training sets. In [22], the unlabeled data is used to change the representation space of the examples, motivated by cases where the class conditional distributions are gaussian. These methods rely on the fact that bipartite ranking data has the form of binary classified data. It is unclear whether these approaches can be extended to more general ranking formulations. In contrast, our multiview method uses a pseudo-labeling step induced by the ranking on the unlabeled data, which should be easier to extend to more general forms of feedback. To the best of our knowledge, all works on semi-supervised ranking have been done in the single view setting. Through the use of multiple views, our approach naturally takes into account the ranking information on the unlabeled set to improve the rankers' performance.

3 Semi-supervised Multiview Learning for Ranking

We present the framework of multiview, semi-supervised ranking with bipartite feedback. We then describe the learning principle underlying our algorithm, presented in Section 4.

3.1 Framework

In bipartite ranking problems, the labeled data take the form of a set $Z = (\mathbf{x}^i, y^i)_{i=1}^n$ of (observation, target) pairs, where $y^i \in \{-1, +1\}$ is called the relevance of observation $\mathbf{x}^i$. Following the standard assumption in machine learning, we assume these examples to be sampled i.i.d. from some fixed (but unknown) distribution, and we denote by (X, Y) a generic pair of random variables which follows that distribution. In a semi-supervised learning setting, we also assume we have access to a pool of unlabeled examples $U = (x^{n+j})_{j=1}^m$ which are i.i.d. and follow the same distribution as X.

In the single-view setting, the goal of bipartite ranking is to learn a function h which assigns a score to any possible input, so that relevant observations (i.e. those with $y = +1$) obtain higher scores than irrelevant ones. The ranking criterion to be optimized is usually taken as the Area Under the ROC Curve (`AUC`). As shown in [9], the goal of learning is then to minimize the ranking risk:

$$L(h) = \mathbb{P}\big((Y - Y')sgn(h(X) - h(X')) < 0\big) \tag{1}$$

where (X', Y') is an independent copy of (X, Y), $sgn(t) = 2\mathtt{I}_{\{t \geq 0\}} - 1$ is the sign function and $\mathtt{I}_{\{.\}}$ is the indicator function. This risk can be estimated on the labeled set by a U-statistics (which is the `AUC`, up to an affine transformation):

$$\hat{L}_Z(h) = \frac{1}{n(n-1)} \sum_{i,j} \mathtt{I}_{\{y_i > y_j\}} \mathtt{I}_{\{h(\mathbf{x}^i) \leq h(\mathbf{x}^j)\}}$$

In multiview learning, an observation (in our case, a multilingual document) $\mathbf{x} = (x_1, ..., x_V)$ is described in several representation spaces $\mathcal{X}_v, v \in \{1 \dots V\}$,

such that each representation (here, a translation in a given language) x_v can be used to build a predictor. Following the framework of [25] for multiview classification or regression, we can define the objective of multiview ranking as jointly learning *view-specific* scoring functions $h_v : \mathcal{X}_v \to \mathbb{R}$ (in our case, h_v only considers the translation of the documents in the v-th language) so that their average risk is small, where the joint learning of these view-specific predictors consists in constraining them to agree with each other (i.e. have similar predictions). Such a principle is amenable to semi-supervised learning since the agreement between predictors can be measured without knowing the labels of the observations, and can thus be estimated (and optimized) from the pool of unlabeled data. Since constraining the view-specific predictors to have a low disagreement reduces the function space, one can then expect better generalization guarantees using semi-supervised multiview learning than using plain supervised learning.

More formally, suppose we are given V view-specific scoring function sets $\mathcal{H}_1, ..., \mathcal{H}_V$ and a *disagreement* function $D : \mathcal{H}_1 \times ... \times \mathcal{H}_V \to [0, 1]$ (the exact definition of D is given in the next subsection). We can then define:

$$\forall t \in [0, 1], \mathcal{H}(t) = \{(h_1, .., h_V) \in \mathcal{H}_1 \times ... \times \mathcal{H}_V : D(h_1, \ldots, h_V) \leq t\} \ , \qquad (2)$$

which is the set of tuples $(h_1, ..., h_V)$ which have a disagreement smaller than $t \in [0, 1]$. Using VC dimension [9] or Rademacher complexity arguments [26] to obtain uniform generalization error bounds for ranking, we can find some function $\mathcal{R}_n(\mathcal{H}(t), \delta)$ which increases with t, such that for any t and $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the random draws of Z, we have:

$$\forall (h_1, ..., h_V) \in \mathcal{H}(t), \frac{1}{V} \sum_{v=1}^{V} L(h_v) \leq \frac{1}{V} \sum_{v=1}^{V} \hat{L}_Z(h_v) + \mathcal{R}_n(\mathcal{H}(t), \delta) \ . \qquad (3)$$

This error bound gives us the principle of semi-supervised, multiview ranking: after an appropriate design of the disagreement function D so that it can be estimated on the pool of unlabeled data, the learning algorithm will aim at optimizing the generalization guarantee Eq. (3) by searching among the view-specific scoring functions with small empirical ranking risk, a tuple $(h_1, ..., h_V)$ with a small empirical disagreement on the pool of unlabeled data.

3.2 Disagreement for Bipartite Ranking

The semi-supervised multiview learning process described above is linked to an appropriate measure of disagreement between view-specific scoring functions. Since the ranking risk (and the `AUC`) linearly decompose into pairwise comparisons between scores, a natural measure of disagreement between two scoring functions h_v and $h_{v'}$ is the probability, over any two random observations, that they do not predict the same ordering:

$$D(h_v, h_{v'}) = \mathbb{P}\big(sgn(h_v(X) - h_v(X')) \neq sgn(h_{v'}(X) - h_{v'}(X'))\big) \ ,$$

which can be estimated on the *unlabeled* data set U by:

$$\widehat{D}_U(h_v, h_{v'}) = \frac{1}{m(m-1)} \sum_{i \neq j} \mathtt{I}_{\left\{ sgn\left(h_v(x_v^{n+i}) - h_v(x_v^{n+j})\right) \neq sgn\left(h_{v'}(x_v^{n+i}) - h_{v'}(x_v^{n+j})\right)\right\}} .$$

We may note that the empirical disagreement is exactly Kendall's tau between the two rankings predicted on U by h_v and $h_{v'}$. This notion of disagreement (and its empirical counterpart) can then be extended to more than two views by taking the average disagreement between scoring functions for any pair of views:

$$D(h_1, \ldots, h_V) = \frac{2}{V(V-1)} \sum_{v<v'} D(h_v, h_{v'}) \text{ and } \widehat{D}_U(h_1, \ldots, h_V) = \frac{2}{V(V-1)} \sum_{v<v'} \widehat{D}_U(h_v, h_{v'}) . \tag{4}$$

Continuing the generalization error bound of Eq. (3), we can note that the empirical disagreement also has the form of a U-statistics, so that VC-dimension or Rademacher arguments can also be used to obtain a uniform (over the whole set of functions $\mathcal{H} = \mathcal{H}_1 \times ... \times \mathcal{H}_V$) bound on $D(h_1, \ldots, h_V) - \widehat{D}_U(h_1, \ldots, h_V)$. Denoting $\mathcal{G}_m(\mathcal{H}, \delta)$ such a bound, we have:

$$\mathbb{P}\big(\sup_{h_v \in \mathcal{H}_v} [D(h_1, \ldots, h_V) - \widehat{D}_U(h_1, \ldots, h_V)] \leq \mathcal{G}_m(\mathcal{H}, \delta) \big) \geq 1 - \delta ,$$

where the probability is taken over U. Using the union bound and plugging this bound into Eq. (3), we have, for any $t \in [0, 1]$, with probability at least $1 - 2\delta$ over both Z and U:

$$\forall (h_1, ..., h_V) \text{ s.t. } \widehat{D}_U(h_1, \ldots, h_V) \leq t,$$
$$\frac{1}{V} \sum_{v=1}^{V} L(h_v) \leq \frac{1}{V} \sum_{v=1}^{V} \hat{L}_Z(h_v) + \mathcal{R}_n(\mathcal{H}(t^*), \delta) , \text{ with } t^* = t + \mathcal{G}_m(\mathcal{H}, \delta) .$$

When the unlabeled dataset is large (which is typically the case in semi-supervised learning), $\mathcal{G}_m(\mathcal{H}, \delta)$ will be small so that the empirical disagreement will be close to the true one. Thus, considering the last error bound, one can see that when there are many empirical risk minimizers in $\mathcal{H}$ (which is typically true when the labeled training set is very small), we may expect much better generalization guarantees for tuples $(h_1, ..., h_V)$ with low disagreement. This is precisely what the algorithm presented in the next section aims at, by iteratively finding view-specific scoring functions with decreasing disagreement (and small empirical risk) using a co-training like procedure, until the disagreement does not improve.

Remark 1 *The authors of [25] argue that the notion of disagreement used in multiview learning should be closely related to the definition of risk, in the sense that they should satisfy a so-called* inverse Lipschitz condition *(see Assumption 2 of [25]). In our case of bipartite ranking, a Bayes-optimal predictor is* $\rho(x) = \mathbb{P}(Y=1 | X=x)$ *[9], and, using our notion of disagreement, the excess risk of any scoring function* h *can be written as* $L(f) - L(\rho) = \mathbb{E}\big[|\rho(X) - \rho(X')| \, D(h, \rho)\big]$. *With a low-noise assumption for ranking similar to the one used by [9] (formally:*

$\exists c > 0, \exists \alpha \in (0,1)$ such that $\mathbb{E}\big[|\rho(X) - \rho(X')|^{-\alpha}\big] \leq c$), we can show that $D(h,\rho) \leq \sqrt{c}\,(L(f) - L(\rho))^{\alpha/2}$, which is precisely an inverse Lipschtz condition of [25]. Thus, in low-noise settings for bipartite ranking, one can obtain strong theoretical results with our notion of disagreement, similar to those of Theorem 2 of [25] (up to a straightforward extension of their framework to ranking).

4 Algorithm

The learning process described above states that we should look for view-specific functions with high `AUC` on the labeled training set, while minimizing the disagreement between the view-specific rankers on the unlabeled dataset.

To that end, we propose an algorithm inspired by pseudo-labeling techniques like iterative co-training [7]. Our approach relies on a supervised learning algorithm for bipartite ranking, and iteratively trains independent rankers on each view with a pseudo-labeling technique: at each round, some unlabeled examples are added to the training set, and their target value is set using the consensus prediction of the view-specific rankers of the previous iteration.

In classification tasks, the pseudo-labeling consists of aggregating the class labels predicted by the view-specific classifiers, for instance taking a majority vote. The unlabeled examples added to the training set at each round are chosen using a measure of confidence in the pseudo-label, in order to avoid adding incorrectly labeled examples to the training material. Although the pseudo-labeling technique used in iterative co-training is not intended to minimize the disagreement between different views, it does empirically tend to decrease the disagreement on the unlabeled set because each classifier is trained with an increasing portion of examples pseudo-labeled by the other classifier. Pseudo-labeling techniques are thus a natural heuristic for learning functions with low disagreement.

Considering our notion of empirical disagreement Eq. (4), it is then natural to define a notion of pseudo-labeling on *pairs* of unlabeled observations: a pair $(\mathbf{x}^{n+i}, \mathbf{x}^{n+j})$ would be labeled $+1$ if the various view-classifiers agree on $h_v(x_v^{n+i}) > h_v(x_v^{n+j})$, and -1 if they agree on the inverse relative ordering. After pseudo-labeling, we would then obtain a training set with pseudo-pairwise preferences (instead of pseudo labels in $\{-1, 1\}$). From a computational point of view, however, this procedure would be extremely costly for two reasons. First, it would require a pass over all pairs of unlabeled inputs at each round. Since there are about m^2 pairs, this is too large by an order of magnitude. Secondly, the pairs of unlabeled inputs selected to be added in the training set do not have the structure of a proper bipartite ranking. The underlying supervised learning algorithm should then be an algorithm that can deal with arbitrary pairwise preferences, which have $\Omega(\ell^2)$ space and time complexity (ℓ is the number of objects in the training set). By contrast, efficient algorithms for bipartite ranking like RankBoost [13] or $\texttt{SVM}^{multi}$ [16] run in time $\tilde{O}(\ell)$ and require $O(\ell)$ space.

Algorithm 1: Semi-supervised Multiview Ranking

`Input:`
▷ supervised bipartite ranking algorithm: $\mathcal{A}$;
▷ size of the random pairs sample: S;
▷ labeled $Z = (\mathbf{x}^i, y^i)_{i=1}^n$, and unlabeled $U = (\mathbf{x}^{n+j})_{j=1}^m$ multiview training data;
`Initialize:`
for each view, train $h_v^{(0)}$ on Z with $\mathcal{A}$.
$t \leftarrow 0$;
repeat
  for $s = 1..S$ **do**
    $(i, j) = \text{sample}\big(\big\{(k, \ell) \in \{1, ..., m\}^2, k \neq \ell\big\}\big)$
    if $\forall v, h_v^{(t)}(x_v^{n+i}) > h_v^{(t)}(x_v^{n+j})$ **then**
      $Z \leftarrow Z \cup \big\{(x^{n+i}, +1), (x^{n+j}, -1)\big\}$
    else if $\forall v, h_v^{(t)}(x_v^{n+i}) < h_v^{(t)}(x_v^{n+j})$ **then**
      $Z \leftarrow Z \cup \big\{(x^{n+i}, -1), (x^{n+j}, +1)\big\}$
    end if
  end for
  $t \leftarrow t + 1$;
  for each view, train $h_v^{(t)}$ on Z with $\mathcal{A}$;
until $\hat{D}_U\big(h_1^{(t)}, ..., h_V^{(t)}\big) \geq \hat{D}_U\big(h_1^{(t-1)}, ..., h_V^{(t-1)}\big)$
`Output:` $\forall v \in \{1, .., V\}, h_v^{(t)}$;

4.1 Weighted Pseudo-labeling

Our multiview approach to semi-supervised bipartite ranking follows existing iterative pseudo-labeling methods for classification, but relies on two ingredients to reduce the overall time and space complexity to $\tilde{O}(n + m)$.

The first one is a reduction from the pseudo-labeled pairs to bipartite ranking in order to use efficient learning to rank algorithms. The second one is a straightforward random sampling of pairs at each iteration rather than considering all possible pairs of unlabeled examples.

The algorithm is fully described in Algorithm 1. In an initialization step, each view-specific ranker is trained independently on the labeled training set. Then, the algorithm iteratively re-trains one ranker per view on increasing training sets composed of the initial labeled examples, and additional pseudo-labeled examples. The first step of each iteration is the pseudo-labeling step, where we increase the size of the labeled training set. Following iterative pseudo-labeling methods for classification (but applied here to pairs of inputs) we pass through unlabeled pairs and decide whether or not they contain information that should be added to the training set based on a measure of confidence in the pseudo-label.

Following [3], we select only the pairs of examples for which all the view-specific rankers agree on the relative ordering. This requirement of unanimity may be too restrictive when there are many views, but we observed that it works very well in practice (see section 5.2). It is not a major point of our algorithm and can be relaxed if too many pairs are ignored in this step.

Once the pairs are selected, we do not add them directly in the training set for computational reasons: bipartite ranking algorithms are very efficient because their implementation makes heavy use of bipartite structure of the preference graph. In order to keep the preference graph bipartite, we actually add binary labeled inputs: for each selected example pair, the input which is scored higher by all view-specific rankers is added to the training set with label $+1$, and the other example in the pair is added with label -1. The crucial point here is that examples may be added several times, possibly with differing labels. Therefore, examples are pseudo-labeled and implicitly weighted in the training set, so that the algorithm will learn to rank the unlabeled examples according to how many pairs they appear in as the greater or lower element.

If this process was applied to the entire unlabeled set, it would require a pass over all pairs of unlabeled inputs, leading to $O(m^2)$ time complexity. In order to avoid this overwhelming cost, we randomly select a much smaller number of pairs (in our experiments, we sample $15,000$ pairs at each iteration, from a set of $60,000$ unlabeled examples).

The iterative procedure is repeated until the disagreement does not decrease after re-training. In order to avoid the costly computation of the disagreement at each iteration, it is estimated using the pairs sampled at the current iteration.

4.2 Supervised ranking algorithm

Our semi-supervised process relies on an underlying efficient algorithm for learning bipartite ranking functions in a fully supervised setting. In this paper, we use a linear `SVM` for ranking, since linear functions with a bag-of-words representation are known to perform very well on textual data.

For each view, v, denoting $Z = (\mathbf{x}^i, y^i)_{i=1}^{\ell}$ the training set available at some given iteration of the algorithm, we learn a linear scoring function h_v of the form $h_v(x) = \langle \mathbf{w}_v, x_v \rangle$ where $\langle ., . \rangle$ is the dot product in Euclidian space, x_v denotes the bag-of-words representation of document $\mathbf{x}$ in the v-th language, and $\mathbf{w}_v$ is the parameter vector to be learnt for view v.

Learning is carried out by minimizing the following pairwise loss for each view (see e.g. [16]):

$$\mathbf{w}_v = \arg\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 + \sum_{i:y^i=1} \sum_{j:y^j=-1} \max\big(0, 1 - \langle \mathbf{w}, x_v^i - x_v^j \rangle\big)$$

The optimization is carried out with an algorithm similar to SVM^{multi} [16], which has time and space complexity in $\tilde{0}(\ell)$. Note that the training set we consider has only binary-labeled pseudo-examples, but some may appear several times with the same or a different pseudo-label as described before.

5 Experimental Results

We illustrate and validate the usefulness of our algorithm on a large collection of documents covering five languages and six categories. We also investigate

Table 1. Number of documents per language (left) and per class (right) in the Reuters RCV1/RCV2 sub-collection considered in our experiments.

Language	# docs source	# docs translated	Total docs
English	18,758	92,982	111,740
French	26,648	85,092	111,740
German	29,953	81,787	111,740
Italian	24,039	87,701	111,740
Spanish	12,342	99,398	111,740

Topic	# docs	(%)
C15	18,816	16.84
CCAT	21,426	19.17
E21	13,701	12.26
ECAT	19,198	17.18
GCAT	19,178	17.16
M11	19,421	17.39

the impact of the number of annotated documents and the imbalance between relevant and irrelevant documents.

We use a publicly available[1] multilingual multiview text categorization corpus extracted from the Reuters RCV1/RCV2 corpus [3], summarized in Table 1. The corpus is originally *comparable* but was made into a parallel, multiview corpus by translating each original document in all other languages. Each of the 111,740 documents is available in 5 views: original language and four translations. The second column in Table 1 indicates the distribution of source languages for our collection. All documents (originals and translations) were indexed using a standard preprocessing chain and are available already indexed.

For each topic, the bipartite ranking problem is to rank documents within this topic above documents belonging to the other topics. The evaluation is carried out with two standard Information Retrieval metrics: the Average Precision (AvP) and Area Under the ROC Curve (AUC) [20]. Each experiment is performed over 10 random splits (labeled training/unlabeled training/test) of the initial collection. The test split always contains 25% of the documents. All labeled/unlabeled/test splits respect the initial topic and language proportions.[2]

5.1 Models

In order to evaluate the benefits of the semi-supervised, multiview approach, and justify our focus on bipartite ranking as opposed to classification, we compare the following five models:

sVR-SVM: fully supervised, single-view ranking. Train monolingual ranking functions on each view, on labeled data only. It corresponds to $h_v^{(0)}$ in Algorithm 1, uses no unlabeled data, and trains independent monolingual rankers.

SsVR-SVM: semi-supervised single-view ranking. Iterative pseudo-labeling approach propagating labels to neighbouring unlabeled examples, as in [2] but using a SVM ranker instead of boosting.

Conc-SR: semi-supervised single-view ranking on concatenated views. Same as the previous model, but operating on concatenated views, instead of independently on each view.

[1] http://multilingreuters.iit.nrc.ca/

[2] With a minimum of 2 positive examples in each labeled training set.

Table 2. AUC and AvP for four competing models, starting from 10 labeled training examples, averaged over 10 random splits and five languages, keeping real class proportions. $^{\downarrow}$ indicates the performance is significantly worse than the best result (in bold). SmVR-SVM is Algorithm 1.

Strategy	C15		CCAT		E21		ECAT		GCAT		M11	
	AUC	AvP	AUC	AvP	AUC	AvP	AUC	AvP	AUC	AvP	AUC	AvP
sVR-SVM	.669$^{\downarrow}$	.329$^{\downarrow}$	.624$^{\downarrow}$	.291$^{\downarrow}$	.621$^{\downarrow}$	.265$^{\downarrow}$	.638$^{\downarrow}$	.283$^{\downarrow}$	.755$^{\downarrow}$	.418$^{\downarrow}$	.811$^{\downarrow}$	.566$^{\downarrow}$
SmVC-SVM	.698$^{\downarrow}$	.334$^{\downarrow}$	.645$^{\downarrow}$	.312$^{\downarrow}$	.652$^{\downarrow}$	.282$^{\downarrow}$	.649$^{\downarrow}$	.294$^{\downarrow}$	.773$^{\downarrow}$	.434$^{\downarrow}$	.821$^{\downarrow}$	.591$^{\downarrow}$
SsVR-SVM	.724$^{\downarrow}$	.416$^{\downarrow}$	.658$^{\downarrow}$	.324$^{\downarrow}$	.665$^{\downarrow}$	.306	.662$^{\downarrow}$	.307$^{\downarrow}$	.802$^{\downarrow}$	.455$^{\downarrow}$	.836$^{\downarrow}$	.620$^{\downarrow}$
Conc-SR	.752$^{\downarrow}$	.438$^{\downarrow}$	.679$^{\downarrow}$	.333$^{\downarrow}$	.672$^{\downarrow}$	**.311**	.671$^{\downarrow}$	.308	.839$^{\downarrow}$	.501$^{\downarrow}$	.875$^{\downarrow}$	.702$^{\downarrow}$
SmVR-SVM	**.805**	**.453**	**.727**	**.353**	**.681**	**.311**	**.694**	**.316**	**.866**	**.532**	**.901**	**.727**

SmVC-SVM: semi-supervised multi-view classification. Classification counterpart to our ranking approach, iteratively labeling examples based on the consensus of classifiers[3] trained on each view [3].

SmVR-SVM: semi-supervised multi-view ranking (this paper). Combines the multiple views available in the training data, using both the labeled and the unlabeled examples as described in Algorithm 1.

All experiments use linear kernels, SVM regularization parameters are set to the default values,[4] and S (inner loop of Algorithm 1) is set to 15000.[5] Performance is reported in terms of average AUC and AvP of the view-specific scoring functions over the five languages.

5.2 Results

We first compare the performance of all the models and investigate the effect of the various aspects of our model. We compute the Average Precision and AUC of the models obtained using 10 labeled training examples, and the unlabeled training examples (for the models that use them). We chose 10 labeled examples in order to study the role of unlabeled data in the regime where very little annotation is available. Table 2 summarizes the results obtained on our six topics by the approaches described above, averaged over five languages and repeated over 10 random training/test splits. Bold face indicates the highest performance, and $\downarrow$ indicates that the performance is significantly worse than the best result, according to a Wilcoxon rank sum test used at $p < 0.01$ [19]. These results show that our approach, SmVR-SVM, consistently and significantly outperforms the four competing models. Unsurprisingly, the simple baseline sVR-SVM yields the lowest performance. The results clearly show that all semi-supervised strategies outperform that baseline, suggesting that the unlabeled data already significantly improves the performance in both AUC and AvP. However, comparing the semi-supervised rankers shows that the multiview (SmVR-SVM) learning

[3] linear SVMs minimizing misclassification error using SVM-Perf [17].

[4] With little annotation, cross-validation proved unreliable to tune hyperparameters.

[5] Increasing this value has almost no influence on the results.

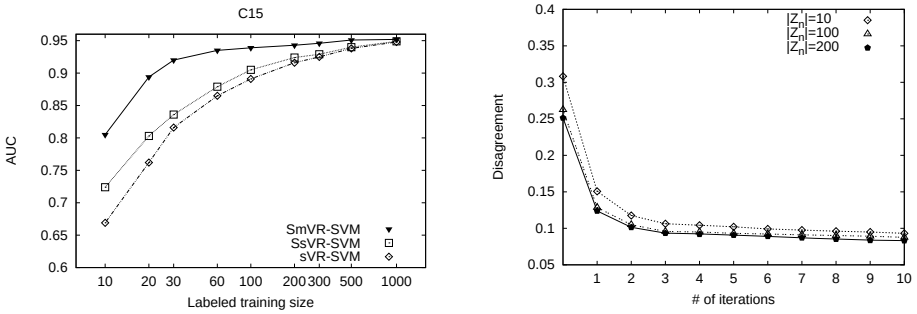

Fig. 1. Left: `AUC` learning curves for topic `C15`, averaged over 10 randomly chosen training/test splits. **Right:** Disagreement, averaged over topics, as training progresses, for 10, 100 and 200 labeled examples.

framework brings more performance increase (up to 16 points in `AvP` for `M11`) than the single view (`SsVR-SVM`) algorithm.

Table 2 also shows that our ranking approach clearly and consistently outperforms semi-supervised multiview classification (`SmVC-SVM`). This suggests that the classification approach is not well suited to solving a bipartite ranking problem even with the help of the richer, multiview information. In fact, this is reinforced by the fact that, in our experiments, even single-view ranking (`SsVR-SVM`) outperforms multiview classification (`SmVC-SVM`). Note also that our approach also outperforms `Conc-SR`, showing that using all views through a simple concatenation is not as efficient as a proper multiview framework.

We now investigate several issues. First, we study the evolution of the performances of our approach depending on the training set size. Then, we show how our approach effectively minimizes the disagreement of the view-specific rankers on the unlabeled data. We also study the influence on the performance of the imbalance of the initial labeled training set, and finally, we investigate the difference between our multiview algorithm and the approach consisting of simply learning on the concatenated views, for increasing numbers of views.

Effect of the Labeled Training Set Size: One of the motivations for using semi-supervised learning is that labeled data is usually costly to acquire. It is therefore of great interest to investigate how the performance of the various algorithms evolve as the number of available labeled examples changes. Figure 1 illustrates this on class `C15` (other classes are qualitatively similar). It shows how the `AUC` evolves with the number of labeled documents in the initial training set. Our experiments correspond to $|Z_n| = 10$, the leftmost points on the curves in the figure. As expected, performance increases monotonically with additional labeled data. The relative ordering observed in Table 2 is maintained throughout the entire range of labeled data, with `SmVR-SVM` performing consistently (but diminishingly) better than `SsVR-SVM`, which in turn does better than `sVR-SVM`.

When there are enough labeled examples, all algorithms actually converge to the same `AUC` value, suggesting that the labeled data carries sufficient informa-

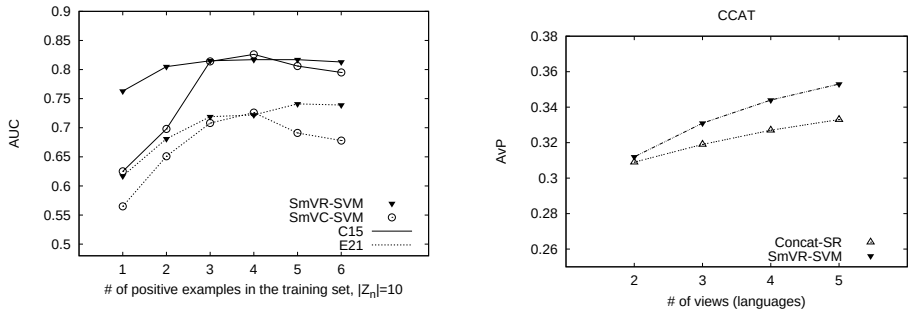

Fig. 2. Left: `AUC` vs. number of positive examples in the 10 labeled training documents, on two topics, for ranking vs. classification. **Right:** `AvP` vs. # of views/languages on topic `CCAT` for multiview learning (`SmVR-SVM`) vs. concatenation (`Conc-SR`).

tion and that no *additional* information could be extracted from the multiview unlabeled examples. For low amounts of labeled training data, however, the contribution of the unlabeled data used in semi-supervised learning is clear.

Evolution of the Disagreement: The motivation of our algorithm is that we improve the generalization performance by minimizing the disagreement between the rankers trained on the different views. Figure 1 shows how the disagreement on unlabeled examples (Eq. 4), averaged over all topics, evolves during training. At iteration 0, the disagreement corresponds to that of `sVR-SVM` (the disagreement of models trained independently on each view without using any unlabeled data). The figure shows that for all three training set sizes pictured, as learning progresses, the disagreement decreases towards a small asymptotic limit. Having more labeled examples helps start with a lower disagreement. However, even for 10 labeled instances, multiview learning brings the disagreement well below that observed for the single-view approach with 20 times more labeled data.

Effect of the Number of Positive Examples: The results in Table 2 use labeled training sets respecting the real class proportions. The learning tasks were thus extremely difficult, since very few positive examples were available. As ranking costs are supposed to be more immune to class imbalance than the misclassification error, we investigate how the performance of the classification versus ranking approaches evolve when more positive examples are available as initial labeled training data. Figure 2 compares the performance of the multiview ranking (`SmVC-SVM`) and classification (`SmVR-SVM`) algorithms for increasing numbers of positive examples. We picked two classes, `C15` and `E21`, which yield differing patterns of results in Table 2: the impact of our method is much larger for `C15` than it is for `E21`. In both cases, Figure 2 shows that as the proportion of positive examples nears 50% (5 positive examples), the classification approach becomes more competitive, while the multiview ranking algorithm appears a lot

more robust to class imbalance. In fact, when the number of positive examples grows past 50% (rightmost edge of the graph, 6 positive and 4 negatives), the performance of the classification approach starts decreasing again.

Comparison to Concatenated Views: The weighted pseudo-labeling step of our algorithm uses a unanimous decision over the views to select examples that should be added to the training set. The unanimous vote is used as a confidence measure, in order to avoid introducing too much noise at each iteration. One may then ask how the performance of the multiview approach evolves depending on the number of available languages? Figure 2 plots the `AvP` observed for topic `CCAT`, as a function of the number of available languages, for our algorithm `SmVR-SVM` and for the semi-supervised single view model which uses the concatenation of the views, `Conc-SR`. Results for less than 5 languages are averaged over all possible subsets of languages. The results show that the performance of `SmVR-SVM` and `Conc-SR` increase as more views are available, with a growing advantage for the multiview approach as the number of languages increases. This confirms that the multiview paradigm offers a better semi-supervised learning principle than the single view learning, and is better able to leverage the additional information available in the different view than simple concatenation of the inputs.

6 Conclusion

We presented an algorithm for bipartite ranking with unlabeled data and multiple views, and showed its empirical performance on a multilingual data collection. The algorithm exhibits better ranking performance than both single-view semi-supervised ranking and multiview classification, in particular when the initial labeled training set is highly unbalanced. Our analysis and algorithms are tailored to bipartite ranking. This allowed us to give experimental comparisons with semi-supervised classification algorithms and existing semi-supervised single view ranking algorithms for bipartite ranking. The results show the importance of optimizing a ranking criterion, as well as the relative performances of single view and multiview approaches.

A direct extension of our work is to examine the possibility of multiview, semi-supervised ranking when the reference ranking information is not bipartite, but take the form of either scores on an ordinal scale, or more generally preference relations. Indeed, even though the weighted pseudo-labeling step is specific to bipartite ranking, the learning principle of Section 3, as well as the method for selecting pairs in the algorithm, extend to more general cases in a straightforward manner. Another direction is to extend our method to search problems, where the goal is to infer rankings on a fixed collection depending on a user query.

References

1. S. Agarwal, T. Graepel, R. Herbrich, S. Har-Peled, and D. Roth. Generalization bounds for the area under the roc curve. *JMLR*, 6:393–425, 2005.

2. M. R. Amini, T. V. Truong, and C. Goutte. A boosting algorithm for learning bipartite ranking functions with partially labeled data. In *SIGIR'08*, 2008.
3. M. R. Amini, N. Usunier, and C. Goutte. Learning from multiple partially observed views – an application to multilingual text categorization. In *NIPS-22*, 2009.
4. F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the SMO algorithm. In *ICML-04*, 2004.
5. M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15:1373–1396, 2003.
6. M. Belkin and P. Niyogi. Semi-supervised learning on riemannian manifolds. *Machine Learning*, 56:209–239, 2004.
7. A. Blum and T. M. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*, pages 92–100, 1998.
8. O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.
9. S. Clémençon, G. Lugosi, and N. Vayatis. Ranking and scoring using empirical risk minimization. In *COLT-05*, pages 1–15, 2005.
10. W. W. Cohen, R. E. Schapire, and Y. Singer. Learning to order things. *Journal of Artificial Intelligence Research*, 10:243–270, May 1999.
11. C. Cortes and M. Mohri. AUC optimization vs. error rate minimization. In *NIPS-16*, 2003.
12. T. Diethe, D. R. Hardoon, and J. Shawe-Taylor. Multiview fisher discriminant analysis. In *NIPS workshop on Learning from Multiple Sources*, 2008.
13. Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *JMLR*, 4:933–969, 2003.
14. S. Hoi and R. Jin. Semi-supervised ensemble ranking. In *Proceedings of the* 23^{rd} *National Conference on Artificial intelligence*, pages 634–639, 2008.
15. T. Joachims. Optimizing search engines using clickthrough data. In *KDD*, 2002.
16. T. Joachims. A support vector method for multivariate performance measures. In *ICML-05*, pages 377–384, 2005.
17. T. Joachims. Training linear svms in linear time. In *KDD-06*, pages 217–226, 2006.
18. S. M. Kakade and D. P. Foster. Multi-view regression via canonical correlation analysis. In *COLT-07*, pages 82–96, 2007.
19. E. Lehmann. *Nonparametric Statistical Methods Based on Ranks*. McGraw-Hill, New York, 1975.
20. C. D. Manning, P. Raghavan, and H. Schtze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.
21. K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39:103–134, 2000.
22. L. Ralaivola. Semi-supervised bipartite ranking with the normalized Rayleigh coefficient. In *ESANN-09*, 2009.
23. S. Robertson and J. Callan. *Routing and filtering*, chapter 5, TREC: Experiment and Evaluation in Information Retrieval, pages 99–121. MIT Press, 2005.
24. D. S. Rosenberg and P. L. Bartlett. The Rademacher complexity of co-regularized kernel classes. *JMLR*, pages 396–403, 2007.
25. K. Sridharan and S. M. Kakade. An information theoretic framework for multi-view learning. In *COLT-08*, pages 403–414, 2008.
26. N. Usunier, M.-R. Amini, and P. Gallinari. A data-dependent generalisation error bound for the AUC. In *In Proceedings of the ICML 2005 Workshop on ROC Analysis in Machine Learning*, 2005.
27. X. Zhu. Semi-supervised learning literature survey. Technical report, Carnegie Mellon University Department of Computer Sciences, 2006.