ESANN'2009 proceedings, European Symposium on Artificial Neural Networks - Advances in Computational Intelligence and Learning. Bruges (Belgium), 22-24 April 2009, d-side publi., ISBN 2-930307-09-9.

A self–training method for learning to rank with unlabeled data

Tuong Vinh TRUONG[†], Massih–Reza AMINI[‡] and Patrick GALLINARI[†]

[†] Université Pierre et Marie Curie
104 avenue du President Kennedy 75016 Paris, France

[‡] National Research Canada
283 Alexandre-Taché Boulevard Gatineau, QC J8X 3X7, Canada

Abstract. This paper presents a new algorithm for bipartite ranking functions trained with partially labeled data. The algorithm is an extension of the self–training paradigm developed under the classification framework. We further propose an efficient and scalable optimization method for training linear models though the approach is general in the sense that it can be applied to any classes of scoring functions. Empirical results on several common image and text corpora over the Area Under the ROC Curve (AUC) and the Average Precision measure show that the use of unlabeled data in the training process leads to improve the performance of baseline supervised ranking functions.

1 Introduction

The ranking task has recently received a large attention from the machine learning and information filtering communities. Indeed, many real-life applications require a ranking of objects instead of their classification. For example, in Information Routing, the system receives a datastream and has to rank these examples according to a user's profile [3]. A current machine learning approach consists in learning a real valued function, which orders relevant examples over irrelevant ones[1]. However to learn such a scoring function it is required to label a large number of examples and constituting such labeled training sets is in general a time–consuming and a expensive task. The semi–supervised learning paradigm has been proposed in classification to alleviate this problem: it deals with using unlabeled data simultaneously with a small labeled training set to improve the results of a classifier.

In this paper we address the issue of learning a linear ranking function from both labeled and unlabeled training sets. In other term we are interested in learning a supervised linear scoring function using the available unlabeled data in the training process. Our approach is based on a very common scheme in semi–supervised learning which is the self–training paradigm [7]. The basic idea here is to iteratively improve a model by generating a feedback through scoring unlabeled data. Specifically, we propose a criterion that measures how likely an unlabeled example can be considered as relevant or irrelevant. Thus it can be

[1] This setting is usually referred to as bipartite ranking.

used to bias the score function by increasing or decreasing the scores of unlabeled examples while respecting the ranking on the labeled dataset. This formulation uses an extended ranking SVM objective function over unlabeled data. Once the model is updated, the algorithm reiterates the procedure until stopping conditions are met.
Following the approach of [6] in supervised learning, we adapt it in the optimization procedure and obtain nice scalability properties. Our experimental results on image (USPS, COIL) and text (RCV1) datasets show the effectiveness of our approach when using unlabeled data in learning the supervised scoring function.

2 Supervised Ranking

In this paper we focus on the bipartite ranking task which is a sub-problem of the ranking paradigm. In the former, the learner is given a set $\mathcal{L} = \{(x_i, y_i)\}_{i=1}^{n}$ of n labeled instances $x_i \in \mathbb{R}^d$ each with a relevance judgment y_i. These relevance judgments take a binary value translating the fact that an example x is either relevant ($y = 1$) or irrelevant ($y = -1$) to a given user interest. In such case, the learning problem can be cast in the search of a real-valued function $h : \mathbb{R}^d \to \mathbb{R}$ that assigns higher scores to relevant instances than to irrelevant ones. Formally it consists in optimizing an upper-bound $R(\mathcal{L})$ on the average number of misordering pairs (x, x') with x a relevant instance and x' an irrelevant one.

$$\mathcal{E}(h, \mathcal{L}) = \frac{1}{|\mathcal{L}_1||\mathcal{L}_{-1}|} \sum_{x \in \mathcal{L}_1} \sum_{x' \in \mathcal{L}_{-1}} [\![h(x) > h(x')]\!]$$

Where $\mathcal{L}_1$ (resp. $\mathcal{L}_{-1}$) denotes the set of relevant (resp. irrelevant) instances in $\mathcal{L}$. Note that $\mathcal{E}(h, \mathcal{L})$ is related to the Area of Under Curve (AUC) since $\mathcal{E}(h, \mathcal{L}) \geq 1 - \text{AUC}$.

3 The Semi–supervised Method

In the semi supervised setting, in addition to a labeled training set the learner is given a large set of unlabeled examples, $\mathcal{U} = \{x_i\}_{i=n}^{n+m}$. The task of learning is hence to find the function h from both training sets $\mathcal{L}$ and $\mathcal{U}$.

3.1 The scoring gap

In order to exploit information from $\mathcal{U}$, we assume in this work that given a model, the highest (resp. lowest) scored instances by the latter are probably relevant (resp. irrelevant) to the user need. In other terms, we assume that an example is likely relevant whenever its score is *closer* to those of positive instances than to negative ones. Let $d_h(S, S')$ be a function which measures the *relative difference* of scores of the elements of S and S' assigned by h. The sign of $d_h(S, S')$ indicates if the scores of elements in S are globally above or below of those in S' and its absolute value denotes how far these scores are. In this paper we set $d_h(S, S') = \frac{1}{|S||S'|} \sum_{x \in S} \sum_{x' \in S'} h(x) - h(x')$ and define scoring gap criteria:

- $\delta_h^+(x) = d_h(\{x\}, \mathcal{L}_{-1})$, (referred to as the relevant scoring gap)
- $\delta_h^-(x) = d_h(\mathcal{L}_1, \{x\})$, (the irrelevant scoring gap)
- $\delta_h(x) = \frac{\min\{\delta_h^+(x), \delta_h^-(x)\}}{d_h(\mathcal{L}_1, \mathcal{L}_{-1})}$, (the scoring gap)

A positive (resp. negative) irrelevant scoring gap means that the score of an example x is globally below (resp above) the scores of relevant examples. The higher is this value, the higher is its irrelevancy. Consequently x is assumed to be irrelevant if $\delta_h^+(x) > \delta_h^-(x)$. In the contrary case, the instance is assumed to be relevant. Finally the scoring gap summarizes the gap between an unlabeled score and the ones of the labeled sets.

3.2 The nibbling algorithm

Our algorithm (Algorithm 1) first initializes the model over the labeled training set $\mathcal{L}$. The output of the learner is used to compute scoring gaps for each example in the unlabeled set and unlabeled instances having a scoring gap above a given threshold are removed from $\mathcal{U}$ and added to a new set of labeled examples $\mathcal{V}$. A new scoring function is then learned optimizing a ranking cost computed separately over labeled datasets $\mathcal{L}$ and $\mathcal{V}$:

$$\widehat{R}_{n+m}(h) = R(\mathcal{L}) + \lambda' R(\mathcal{V}) + \lambda \|w\|^2 \tag{1}$$

Where R is an upper-bound over $\mathcal{E}$ and λ' a discount factor that controls the influence of unlabeled data in the learning process. Finally the procedure is iterated and the threshold ζ may gradually decrease until a limit.

In this paper, we consider a linear score function, i.e. $h(x) = w.x$ and the hinge

Algorithm 1 Skeleton of *Nibbling* algorithm

```
Require: a labeled set $\mathcal{L}$ and an unlabeled set $\mathcal{U}$
$\mathcal{V} \leftarrow \emptyset$, $\mathcal{N} \leftarrow \mathcal{U}$
Train a ranking function on $\mathcal{L}$
repeat
  for each example $x$ in $\mathcal{N}$ do
    if $\delta_h(x) < \zeta$ then
      $\mathcal{V}$.APPEND $((x, .))$, $\mathcal{N}$.DELETE$(x)$
    end if
  end for
  Assign a relevance judgment to the newly added instances in $\mathcal{V}$
  Update the model by minimizing (1)
  if no example added in $\mathcal{V}$ then
    $\zeta \leftarrow$ decrease$(\zeta)$
  end if
until $\mathcal{N} = \{\}$ or $\zeta > \zeta_L$
Ensure: the model
```

loss as a convex surrogate of the 0–1 loss. It yields to the following objective function: $R(\mathcal{L}) = \sum_{x \in \mathcal{L}_1} \sum_{x' \in \mathcal{L}_{-1}} \max(0, 1 - w.(x - x'))$. We employ a bundle method [6] for its optimization. In this case the solver was extended to minimize R on $\mathcal{L}$ and $\mathcal{N}$ simultaneous. The optimization procedure is scalable and converges with an ϵ precision [5] in $O(1/\epsilon)$ for any convex but possibly non differentiable objective function. At each iteration, the function and its gradient can be computed in linear time, once the instances are sorted according to their scores.

4 Empirical study

4.1 Experimental setup

We conducted series of experiments on widely used benchmarks modified for semi-supervised learning. These datasets are listed in table 1 and are mostly employed in classification. In our experiments we derived several information routing tasks from these benchmarks by supposing that each class from each collection represents a predefined user need and hence each of its examples are relevant to the information associated to the class. Examples from other classes are supposed to be irrelevant to this class. In our experiments, the RCV1 collection [4] is based on a binary version of the text categorization dataset. In the binary version of this collection, CCAT, ECAT categories from the initial dataset are considered as relevant and GCAT and MCAT as irrelevant. For USPS and COIL[2] we successively consider the 5 first classes as relevant to derivate different information routing tasks.

In order to evaluate the contribution of unlabeled data in the learning phase, we compared our approach (Algorithm 1) with the fully supervised bundle technique [6] maximizing the AUC on the labeled dataset. For evaluation we used both the AUC and Average Precision measures.

dataset	c	d	$n+m$	test set size	positive class ratio
USPS	5	256	7291	2007	0.1
COIL	5	1024	1440	1000	0.05
RCV1-BINARY	2	47236	20242	677399	0.5

Table 1: Datasets properties: c represents the number of topics or user needs, $n + m$ is the number of labeled and unlabeled examples in the training set and d is the dimension of the problem

We further conducted different experiments by varying the parameter λ over the interval $[10^{-2}, 10^{2}]$, λ' over the set $\{10^{-4}, 10^{-2}, 1\}$ and ζ_L over $\{0.1, \ldots, 0.5\}$. The thresehold ζ takes its value in $\{0.01, 0.05, ..., 0.5, 0.6, \ldots, 1\}$. For the supervised method, we vary λ over the set $\{1e^{-4}, 1e^{-2}, \ldots, 1e^{4}\}$. Each experiment is repeated 10 times by randomly choosing the labeled, unlabeled and test sets

[2] http://www.kyb.tuebingen.mpg.de/bs/people/chapelle/lds/

AUC (%)		
Topic	supervised	Algorithm 1
1	88.7	**94.7**$^{\uparrow}$
2	99.7	99.7
3	90.5	**95**$^{\uparrow}$
4	89.5	**91.6**
5	87.1	**92.1**$^{\uparrow}$

Average precision (%)		
Topic	supervised	Algorithm 1
1	76.3	**82.7**
2	99.1	**99.2**
3	69.0	**83.8**$^{\uparrow}$
4	59.9	58.7
5	51.0	**63.7**$^{\uparrow}$

Table 2: AUC and Average precision obtained on USPS.

AUC (%)		
Topic	supervised	Algorithm 1
1	92.2	**96.7**$^{\uparrow}$
2	64.7	64.8
3	87.5	88.0
4	96.2	**97**$^{\uparrow}$
5	74.8	**77.9**$^{\uparrow}$

Average precision (%)		
Topic	supervised	Algorithm 1
1	68.6	**72.1**
2	38.5	35.1
3	33.6	**38.9**
4	74.4	73.2
5	41.3	36.1

Table 3: AUC and Average precision obtained on COIL.

on the initial collection. We set n at 10 for the images datasets. Following [1] the reported results are the best we obtained on the test set for a given pair of parameters (λ, λ'). This protocol allows to estimate the potential of our algorihtm. We further performed a one tail Wilcoxon signed–rank test by comparing the results of both approaches and statistically significant improvements with a precision level of 95% are indicated using the symbol $\uparrow$.

In all cases the use of unlabeled data does not decrease the performance of the supervised approach and in many cases the semi–supervised algorithm is significantly better than the supervised technique. It is also to be noted that even a modified AUC criteria is used in Algorithm 1, the latter still significantly outperforms the supervised model on Average Precision in most cases. This result is of interest as it has been shown that in the supervised case, the optimization of AUC may lead to a suboptimal solution for the average precision measure [2].

AUC (%)		
n	supervised	Algorithm 1
100	94.3	**95.1**$^{\uparrow}$
200	95.8	**97.1**$^{\uparrow}$
400	97.3	**97.8**$^{\uparrow}$

Average precision (%)		
n	supervised	Algorithm 1
100	94.9	**95.8**$^{\uparrow}$
200	96.2	**97.5**$^{\uparrow}$
400	97.5	**98.0**$^{\uparrow}$

Table 4: Results on the binary version of RCV1 for different sizes of the labeled training set.

5 Conclusion

In this paper, we proposed a new semi–supervised algorithm for bipartite ranking. Much work has been done in the design of semi–supervised algorithms under the classification framework and learning to rank with partially labeled data has just been considered recently. The proposed approach is based on the self–training paradigm, as the output of a ranker is iteratively used to select and label unlabeled examples for training a new ranker. Our experimental results make empirical evidence that in many cases unlabeled data can help to learn a more performant ranker than the one learned using only the labeled data and that in the worst case, they do not lead to a decrease of the basic supervised performance. Further, our algorithm can be extended to non–linear scoring functions and other ranking measures can be optimized using the bundle optimization technique.

References

[1] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning.* MIT Press, Cambridge, MA, 2006.

[2] Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 233–240, New York, NY, USA, 2006. ACM.

[3] Raj D. Iyer, David D. Lewis, Robert E. Schapire, Yoram Singer, and Amit Singhal. Boosting for document routing. In *CIKM '00: Proceedings of the ninth international conference on Information and knowledge management*, pages 70–77, New York, NY, USA, 2000. ACM.

[4] David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. Rcv1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5:361–397, 2004.

[5] Alex Smola, S V N Vishwanathan, and Quoc Le. Bundle methods for machine learning. In *Advances in Neural Information Processing Systems 20*, pages 1377–1384. MIT Press, Cambridge, MA, 2008.

[6] Choon Hui Teo, Alex Smola, S. V.N. Vishwanathan, and Quoc Viet Le. A scalable modular convex solver for regularized risk minimization. In *KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 727–736, New York, NY, USA, 2007. ACM.

[7] Jean-Noël Vittaut, Massih-Reza Amini, and Patrick Gallinari. Learning classification with both labeled and unlabeled data. In *Proceedings of the 13th European Conference on Machine Learning (ECML'02)*, pages 468–476, 2002.