Top-rank convolutional neural network and its application to medical image-based diagnosis

Yan Zheng[a,*], Yuchen Zheng[c], Daiki Suehiro[a,b], Seiichi Uchida[a]

[a] *Department of Advanced Information Technology, Kyushu University, Fukuoka, Japan*
[b] *RIKEN, Japan*
[c] *College of Information Science and Technology, Shihezi University, Shihezi, China*

Abstract

Top-rank learning identifies a real-valued ranking function that will provide more *absolute top* samples. These are highly reliable positive samples that are ranked higher than the highest-ranked negative samples. Therefore, top-rank learning is useful for tasks that require reliable decisions. Additionally, it inherits the merits of the ranking functions, such as robustness to the unbalanced condition. However, conventional top-rank learning tasks are formulated as linear or kernel-based problems and are thus limited in coping with complicated tasks. In this study, we propose a Top-rank convolutional neural network (TopRank CNN) to realize top-rank learning with representation learning for complicated tasks. Given that the original objective function of top-rank learning suffers from overfitting, we employ the p-norm relaxation of the original loss function in the proposed method. We prove the usefulness of TopRank CNN experimentally with medical diagnosis tasks that require reliable decisions and robustness to the unbalanced condition.

Keywords: Top-rank learning, Representation learning, Medical diagnosis

*Corresponding author
Email addresses: yan.zheng@human.ait.kyushu-u.ac.jp (Yan Zheng), ouczyc@outlook.com (Yuchen Zheng), suehiro@ait.kyushu-u.ac.jp (Daiki Suehiro), uchida@ait.kyushu-u.ac.jp (Seiichi Uchida)

1. Introduction

Learning to rank is a machine-learning task based on which values are ordered using a ranking function $r(\mathbf{x})$ for each sample $\mathbf{x}$. If $r(\mathbf{x}) > r(\mathbf{x}')$, the sample $\mathbf{x}$ is ranked higher than $\mathbf{x}'$ based on the ranking function r. This approach is extensively used for recommendation systems in which goods or services with higher rankings are recommended.

Bipartite ranking [1] is one of the popular tasks in these types of applications, as outlined in Fig. 1 (a). Given a positive sample set $\Omega^+ = \{\mathbf{x}_1^+, \ldots, \mathbf{x}_m^+\}$ and a negative sample set $\Omega^- = \{\mathbf{x}_1^-, \ldots, \mathbf{x}_n^-\}$, the bipartite ranking task optimizes the ranking function $r(\mathbf{x})$ so that it gives higher ranks to positive samples in Ω^+ than negative samples in Ω^-. Note that Fig. 1 (a) depicts a linear ranking function and the samples on a dotted line have the same rank value, such as r_3.

The bipartite ranking approach has several useful characteristics for improving the unbalanced learning tasks. First, the optimization of bipartite ranking function is equivalent to the maximization of the area-under-the-curve (AUC) [2], thereby providing optimal performance for *detecting* positive samples among all samples $\Omega^+ \cup \Omega^-$ by pushing positive samples with higher ranking scores. Second, the bipartite ranking function is robust to unbalanced data [3]. As it is generally optimized via consideration of all $m \times n$ pairs of positive and negative samples, the difference between the numbers of positive and negative samples becomes insignificant.

Top-rank learning is a version of bipartite ranking [4–6]. Fig. 1 (b) illustrates the concept of top-rank learning. The main difference from bipartite ranking is that top-rank learning attempts to maximize the number of *absolute top* samples, instead of maximizing the AUC. *Absolute top* refers to the positive samples ranked higher than the top-ranked (highest-ranked) negative samples. Therefore, *absolute top* samples are highly reliable positive samples and are clearly different from the negative samples. *Pos@Top* is the ratio of the *absolute top* samples to all the positive samples. Considering that the receiver operating characteristic (ROC) of a ranking task is drawn by changing the rank value as

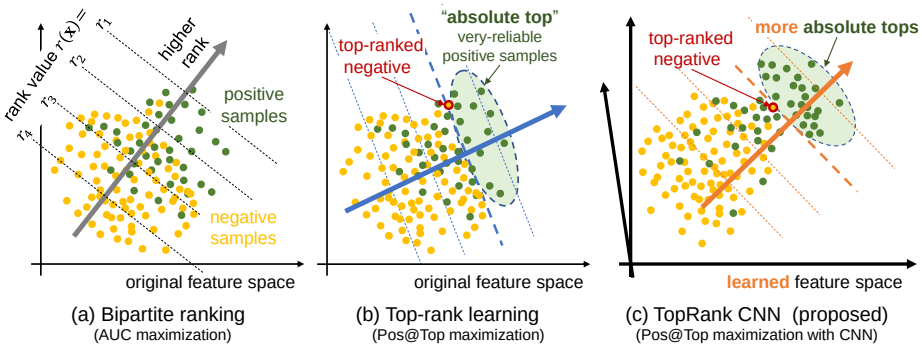


Figure 1: Overview of (a) bipartite ranking, (b) top-rank learning and (c) TopRank CNN (the proposed method).

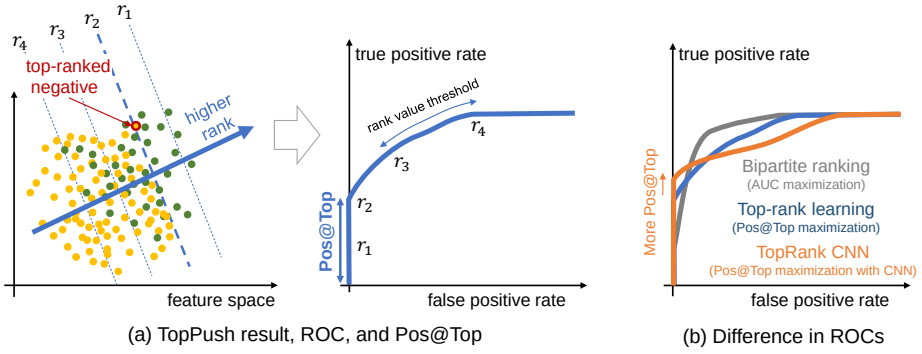


Figure 2: Relationship between the ranking functions and their ROC.

the threshold, Pos@Top is observed as the length of the initial (i.e., leftmost) vertical part of the ROC, as shown in Fig. 2 (a). Consequently, the objective of top-rank learning is to obtain an ROC curve with *not* the maximum AUC *but* with the maximum Pos@Top.

The maximization of Pos@Top is often favorable compared with the maximization of AUC, especially for medical image diagnosis. Assume that positive and negative samples are images of cancerous and healthy tissues, respectively. If a test image is classified as a *absolute top* sample (that is, its rank value is larger than the top-ranked negative sample), medical experts can skip the manual inspection of the test image because they (clearly) depict cancerous tis-

sue [1]. Therefore, if we have a larger Pos@Top (i.e., more *absolute top* samples), their manual inspection efforts can be reduced drastically. Note that top-rank learning inherits the robustness for unbalanced data. Accordingly, this property provides another merit related to the use of top-rank learning for medical diagnosis.

Conventional top-rank learning tasks have been formulated in the forms of linear or kernel-based problems that can be solved by various algorithms, such as the support vector machine (SVM)-based algorithm called TopPush [6]. Unfortunately, this formulation cannot fully utilize the favorable properties of top-rank learning. When positive and negative samples cannot be easily separated, it will be difficult to have a large Pos@Top. Although the use of a kernel function will increase the separability, it is hard to find an appropriate kernel function for the task.

This study proposes the *Top-rank convolutional neural network* (TopRank CNN) that enables top-rank learning with representation learning. If the feature space is optimized for top-rank learning by CNN-based representation learning, we will have more *absolute top* samples, i.e., a larger Pos@Top, as shown in Fig. 1 (c) and Fig. 2 (b). This will enhance the usefulness of top-rank learning for medical image diagnosis and various applications that need reliable decisions.

It should be emphasized that combining the ideas of top-rank learning and CNN is not straightforward owing to the following two points. First, if we use the original objective function of the top-rank learning problem for TopRank CNN, it easily causes heavy-overfitted results to the training data. This is because the feature representation by CNN tries to maximize the Pos@Top of the training data. Therefore, we introduce the p-norm relaxation of the original objective function to avoid the overfitting problem. The experimental results will prove that the TopRank CNN can achieve larger Pos@Top values

[1] Conversely, it is possible to assume that positive and negative samples denote images of normal and cancerous tissues, respectively. In this case, a test image classified as the *absolute top* is "clearly normal" and the medical experts can also ignore it.

under this relaxation with an appropriate hyper-parameter, p. Second, the organization of each minibatch requires more consideration than ordinary CNN training tasks. Given that the loss function is calculated by positive-negative pairs (like bipartite ranking), it is important to ensure that an appropriate number of positive and negative samples are included in each minibatch during the training of TopRank CNN.

The main contributions of this study are as follows:

- To the authors' best knowledge, this is the first proposal of combining top-rank learning with representation learning for medical image analysis by end-to-end way.
- The p-norm relaxation of the loss function enables an end-to-end training framework of top-rank learning and representation learning. This will further enhance the usefulness of top-rank learning.
- Experimental results of medical image-based diagnosis tasks proved that the proposed TopRank CNN can achieve more *Pos@Top* than other methods; this means that TopRank CNN can find more highly reliable positive samples automatically and it will reduce the manual inspection efforts of medical experts.

2. Related work

This section reviews related studies on relevant ranking methods, and especially top-rank learning methods. Studies on ranking methods with deep neural networks are also reviewed.

2.1. Learning to Rank

Techniques for learning to rank have been developed and applied to information retrieval in recent decades [7–10]. Among these, the bipartite ranking method is employed to rank positive samples higher than negative ones [1, 11, 12] in the case that it is possible to separate samples into positive and negative

classes. This method has interesting characteristics that are favorable for various applications, such as document retrieval, recommendation systems, and medical image analysis. For example, Hanley and McNeil theoretically proved that the bipartite ranking is equivalent to AUC maximization [2, 13, 14].

One of the strong merits of learning to rank is its theoretical robustness to *unbalanced data* classification, which is an important classification task in many applications [15–21]. Very often, heuristic approaches, such as undersampling, oversampling, and data synthesis, are employed to deal with data imbalance problems, but they do not have any theoretical support.

In contrast, as proved in [22], RankSVM [7] is robust to class imbalance (because of its ability to maximize AUC). Cruz et al. [23] experimentally confirmed this merit. Nowadays, learning to rank employs unbalanced data tasks, such as information retrieval, signature verification, etc [24–28].

2.2. Top-rank learning

As noted in Section 1, top-rank learning focuses on highly ranked samples. Interestingly, there are several variations in its formulation based on the use of different criteria [29–34]. For example, Joachims [29] proposed a ranking method that maximizes the precision of the top k-ranked samples *precision@k*. Narasimhan and Agarwal [30] maximized partial AUC, which evaluates the AUC between a certain ROC interval. In [31], another formulation that focused on the top-ranked samples was found. Despite the theoretical interests of these formulations, they result in non-convex optimization problems that cannot be solved efficiently.

According to the variations of the formulations, the algorithms for top-rank learning also exhibit variations. For example, Rudin proposed the p-norm push algorithm [35]. Agarwal et al. [36] proposed the InfinitePush algorithm which maximize the number of absolute top samples. Li et al. [6] also proposed the TopPush algorithm maximizing the number of *absolute top* samples more efficiently than InfinitePush.

2.3. Ranking with deep neural networks

With the development of deep learning, recent studies have applied the idea of learning to rank to a variety of applications, such as information retrieval [37–39], face recognition [40, 41], and person re-identification [42, 43].

The typical motivation related to the combination of deep learning and learning to rank is to realize a metric learning method, whereby the similarity is evaluated by a ranking function. For example, Huang et al. [37] proposed the deep structure semantic model (DSSM), and Shen et al. [44] proposed the convolutional deep structured semantic Model (CDSSM) to directly apply deep neural networks to obtain the semantic representations of queries and documents. The ranking score was produced by computing their cosine similarity and achieved good performance for web searching. More recently, Pang et al. [45] proposed DeepRank to capture important semantic features for information retrieval. The relevance scores of DeepRank are obtained by using the pairwise ranking loss. DeepRank achieved higher state-of-the-art results compared with traditional learning to rank methods used for information retrieval.

In [43], Chen et al. also proposed an effective deep ranking framework combining learning to rank algorithm with deep CNN for solving person re-identification tasks. They aimed to obtain a similarity metric in which the top-k closest candidate images to the query is matched with the query. To obtain accurate results at the top, they applied learning-to-rank techniques with metric learning and CNN.

In these approaches, CNN-based representation learning played an important role in enhancing the effectiveness of traditional learning to rank methods. Furthermore, it offered a feature space wherein the positive samples were pushed to be ranked before any negative samples by maximizing the AUC.

Considering the effectiveness of the top-rank learning method on optimizing the accuracy at the top, some studies combined the idea of top-rank learning (by optimizing different criteria, e.g. the precision at the top, the number of absolute tops) with deep learning models [46, 47].

In [46], Song et al. employed top-rank learning to solve the binary coding

problem for image data. The target of this work was to learn coding functions that the data in the same group were close to each other in the coding space (Hamming space), which focused on the closeness of the top position (i.e., representative data in the same group) in the Hamming space, and optimized the precision of the top position.

To learn non-linear representation for ranking, Geng et al. used the concept of pos@top to obtain discriminative local features in data [47], which achieved better performance on pos@top. However, this method just used the pos@top-based loss to learn CNN-based representation, not for the classifier. More importantly, this method was designed to solve the multiple-instance problem and hard to solve vectorized data that cannot be split into a set of instances.

In this study, we broken through the limitations of [47] and proposed TopRank CNN to integrate the top-rank learning and representation learning using p-norm relaxation by an end-to-end framework. TopRank CNN supported the mapping of samples to the rankable deep feature space and maximized the number of positive samples ranked before the top-ranked negative samples. In this work, the application of TopRank CNN on medical image diagnosis produced a better performance than the comparative methods.

3. Top-rank convolutional neural network

3.1. Objective function to maximize Pos@Top

The typical bipartite ranking problem needs to determine a real-valued ranking function $r(\mathbf{x})$ that yields higher scores for positive samples compared with negative samples. Let $\Omega = \Omega^+ \cup \Omega^-$ be the training set and $\Omega^+ = \{\mathbf{x}_1^+, \ldots\ldots, \mathbf{x}_m^+\}$ and $\Omega^- = \{\mathbf{x}_1^-, \ldots\ldots, \mathbf{x}_n^-\}$ represent the positive sample set and the negative sample set, respectively. The objective of the typical bipartite ranking problem is the maximization of AUC, which is defined as

$$\mathrm{AUC} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} I(r(\mathbf{x}_i^+) > r(\mathbf{x}_j^-)), \tag{1}$$

where $I(\cdot)$ is the indicator function. Consequently, the bipartite ranking problem is formulated as the minimization problem of the loss function $\mathcal{J}_{\text{bipartite}}$ with respect to $r(\mathbf{x})$:

$$\mathcal{J}_{\text{bipartite}} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} I(r(\mathbf{x}_i^+) \leq r(\mathbf{x}_j^-)). \tag{2}$$

The objective of our TopRank CNN is not to maximize AUC but to maximize Pos@Top. Therefore, we need to use a different objective function. As shown in Figs. 2(a) and (b), Pos@Top is the ratio of absolute top samples among all m positive samples. Given that the absolute top samples are the "most reliable positive" (or "very positive") samples, maximizing Pos@Top is beneficial to many applications. Formally, Pos@Top is defined as

$$\text{Pos@Top} = \frac{1}{m} \sum_{i=1}^{m} I(r(\mathbf{x}_i^+) > \max_{1 \leq j \leq n} r(\mathbf{x}_j^-)), \tag{3}$$

where $\max_{1 \leq j \leq n} r(\mathbf{x}_j^-)$ denotes the ranking score of top-ranked negative samples. We replace the indicator function $I(\cdot)$ by a surrogate function $\ell(\cdot)$, which is a convex, non-decreasing, and differentiable function. Thus, the loss function to be minimized is formulated as follows,

$$\begin{aligned} \mathcal{J}_{\text{Pos@Top}} &= \frac{1}{m} \sum_{i=1}^{m} \ell \left(\max_{1 \leq j \leq n} r(\mathbf{x}_j^-) - r(\mathbf{x}_i^+) \right) \\ &= \frac{1}{m} \sum_{i=1}^{m} \max_{1 \leq j \leq n} \ell \left(r(\mathbf{x}_j^-) - r(\mathbf{x}_i^+) \right). \end{aligned} \tag{4}$$

In the later experiments, $\ell(z) = \log(1 + e^{-z})$ is used as the surrogate function. It should be emphasized again that the minimization of (4) does not maximize AUC but *Pos@Top*. Therefore, the ranking function by (4) may give a smaller AUC than (2).

3.2. p-norm relaxation

Although we can use Eq. (4) for the loss function of CNN, there is a large risk of overfitting to the training data. CNN has a powerful representation learning ability so as to minimize the loss function. In an extreme case with Eq. (4), all

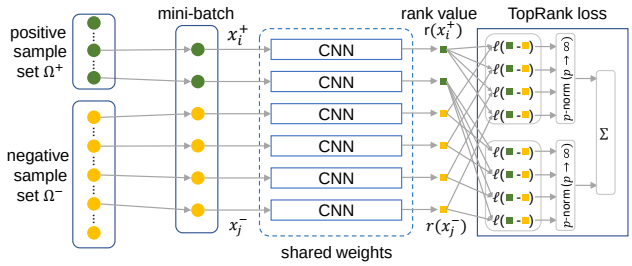


Figure 3: The architecture of TopRank CNN.

the positive samples become the absolute top samples (by making them very different from the negative samples) and thus achieves *Pos@Top*=100%. Considering the presence of outliers in the training samples, this "perfect" situation does not often work in the case of the test samples.

Therefore, we introduce the p-norm relaxation of the max operation in Eq.(4). Specifically, we use the following loss function, called TopRank loss,

$$\mathcal{J}_{\text{TopRankCNN}} = \frac{1}{m}\sum_{i=1}^{m}\left(\sum_{j=1}^{n}\left(\ell(r(\mathbf{x}_i^+) - r(\mathbf{x}_j^-))\right)^p\right)^{\frac{1}{p}}. \tag{5}$$

If the hyper-parameter $p \to \infty$, $\mathcal{J}_{\text{TopRankCNN}} \to \mathcal{J}_{\text{Pos@Top}}$. If we set p at a smaller value, say 16 or 32, our loss function $\mathcal{J}_{\text{TopRankCNN}}$ has a milder effect on the maximization of Pos@Top, and can avoid the overfitting issue. This effect will be proved in a subsequent experiment.

3.3. TopRank CNN

Fig. 3 shows the network architecture of the proposed TopRank CNN that is designed to integrate top-rank learning and representation learning using an end-to-end framework with the TopRank loss (5). Based on this architecture, we will have an appropriate feature representation for higher Pos@Top (i.e., more absolute tops), as shown in Fig. 1(c).

Given that the TopRank loss function needs positive and negative pairs, we used Siamese architecture [48], which consists of two identical CNNs with the same weights, and used two sample inputs. It should be noted that our

TopRank CNN is still very different from typical Siamese networks given its purpose to derive the ranking function $r(\mathbf{x})$ with the TopRank loss (5) for the maximization of *Pos@Top*.

One technical topic that is unique to TopRank CNN is the organization of samples in each minibatch. For each minibatch, the TopRank loss (5) is calculated by using all the pairs of positive and negative samples. Thus, each minibatch needs to contain the samples from both classes (if a minibatch is comprised of the samples from a single class, it is impossible to calculate the loss). More importantly, (5) is the p-norm relaxation of (4), and thus its minimization will push the positive samples to have higher ranks than *the top-ranked negative samples within the minibatch*. This means that it is implicitly expected that each minibatch includes a negative sample that is close to the top-ranked negative samples. Consequently, each minibatch is better organized by using more negative samples than positive samples to satisfy the expectation as much as possible.

The value of the hyper-parameter p will be set by using a validation set. In general, a very large p should be avoided, although it makes (5) similar to its original form (4). This is because large p values easily cause numerical overflows (scaling techniques for normalizing the input data by dividing its maximum value may avoid overflows with larger p values, but it may result in underflow). Thus, in practice, we sometimes need to stop the validation steps before a larger p value causes an overflow.

4. Experiment on an unbalanced CIFAR-10 dataset

In the experiments, the comparative models and the proposed TopRank CNN model were implemented using the platform of TensorFlow and the Python 3.5.2 packages. The training and test model was based on the GPU environment of two NVIDIA GeForce GTX 1080 Ti GPU with 10GB RAM.

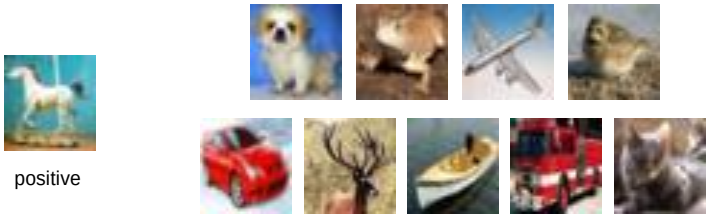


Figure 4: Examples of positive and negative classes from the unbalanced CIFAR-10 dataset.

4.1. Dataset

To investigate the basic performance of TopRank CNN for solving unbalanced problems, we first applied it to an unbalanced version of the CIFAR-10 dataset [49]. In the unbalanced CIFAR-10, the class "horses" was selected as the positive class, and the other nine classes were negative class, as shown in Fig. 4 (note that the selection of "horses" was random). From the training set of CIFAR-10, we randomly selected 1000 positive and 9000 negative samples to build the validation set. Thus, the training, validation, and testing sets contain 4000, 1000, and 1000 positive samples and 36000, 9000, and 9000 negative samples, respectively. Each sample corresponds to a $32 \times 32 \times 3$ color image.

4.2. Comparative methods

The following experiments were conducted with five comparative methods.

- $\text{TopPush}_{\text{CNN}}$: The original TopPush algorithm [6] used the feature extracted from the previous fully connected layer before the final layer of the following CNN_{soft}.
- $\text{RankSVM}_{\text{CNN}}$: The RankSVM algorithm [7] also used the feature extracted from the previous fully connected layer before the final layer of the following CNN_{soft}.
- CNN_{soft}: CNN trained with the standard softmax loss for binary classification (positive and negative classes).

- $\mathrm{CNN}_{\mathrm{focal}}$: CNN was trained with a focal loss [20] and is one of the most extensively known solutions used to cope with the data-imbalance binary classification problem.

- $\mathrm{CNN}_{\mathrm{pos@top}}$: CNN trained with $\mathcal{J}_{\mathrm{Pos@Top}}$ of Eq. (4), instead of its p-norm relaxation.

Note that we used the same architecture for all the CNN-based methods except for the final loss layer.

4.3. TopRank CNN architecture and hyper-parameters

As shown in Fig. 3, TopRank CNN is composed of two identical CNNs (for a positive-negative samples pair of input) that share the same weights. Each CNN consists of three convolutional layers and two fully connected layers, namely, the rectified linear unit (ReLU) activation function and the max-pooling layers that followed each convolutional layer. The learning rate decay method was employed to adjust the learning rate during the training, with an initial learning rate of 0.01, and was updated with a decay factor of 0.90 after every epoch. Each minibatch contained 5 positive and 450 negative samples — this unbalance minibatch is mandatory owing to the reason outlined in Section 3.3. The stochastic gradient descent (SGD) algorithm was used to optimize the network.

The validation set was used to determine the hyper-parameter of $p \in \{2, 4, 8, 16, 32, 64\}$ for TopRank CNN, $\lambda \in \{10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}\}$ for $\mathrm{TopPush}_{\mathrm{CNN}}$, and $C \in \{10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}\}$ for $\mathrm{RankSVM}_{\mathrm{CNN}}$. Note that a larger p causes numerical overflow. Thus, p cannot be larger than 64. The hyper-parameter of class weight $c = m/n$ was also utilized to balance the positive and negative classes for the training of conventional CNN.

4.4. Results on the unbalanced CIFAR-10 dataset

Table 1 shows the AUC and Pos@Top of the proposed method and the five comparative methods. In the cases of the classification methods, such as $\mathrm{CNN}_{\mathrm{soft}}$, $\mathrm{CNN}_{\mathrm{focal}}$, we used their soft-max values to calculate AUC and Pos@Top. This table indicates the following facts:

Table 1: *Pos@Top* and AUC on the test set of the unbalanced CIFAR-10.

Method	Feature	Algorithm	Loss	Hyper-param.	AUC Test	Pos@Top Train	Pos@Top Test
$\text{TopPush}_{\text{CNN}}$	CNN_{soft}	TopPush	Eq.(4)	$\lambda = 10$	0.7186	0.0383	0.0270
$\text{RankSVM}_{\text{CNN}}$	CNN_{soft}	RankSVM	Eq.(2)	$C = 0.01$	0.8912	0.0475	0.0210
CNN_{soft}	end-to-end	CNN	Softmax	-	0.8804	0.0410	0.0260
$\text{CNN}_{\text{focal}}$	end-to-end	CNN	Focal	-	0.9124	0.0318	0.0590
$\text{CNN}_{\text{pos@top}}$	end-to-end	CNN	Eq.(4)	-	0.8894	0.4635	0.0990
TopRank CNN (ours)	end-to-end	CNN	Eq.(5)	$p = 16$	**0.9465**	0.4910	**0.1950**

1) Among all the methods, the proposed TopRank CNN achieved the best performance on Pos@Top, as expected
2) TopRank CNN obtained a higher Pos@Top than the $\text{TopPush}_{\text{CNN}}$ that used a fixed feature representation given by CNN_{soft}. This revealed the superiority of TopRank CNN on the maximization of Pos@Top by end-to-end learning
3) TopRank CNN also achieved the best AUC, although its main objective was not to maximize AUC but Pos@Top. This suggests that the feature representation that boosts Pos@Top gives a positive effect to maximize AUC. In fact, TopRank CNN achieves higher AUC values than $\text{RankSVM}_{\text{CNN}}$ (which just tries to maximize AUC)
4) Comparing TopRank CNN to $\text{CNN}_{\text{pos@top}}$, p-norm relaxation by (5) is effective in reducing the overfitting to the training samples. In fact, TopRank CNN achieves a similar Pos@Top (0.0960) on the test samples when p is increased to 64
5) $\text{CNN}_{\text{focal}}$ achieves a better performance than CNN_{soft} and thus shows its strength in the unbalanced task; however, its AUC, as well as Pos@Top does not exceed TopRank CNN

To observe the ability of TopRank CNN to maximize Pos@Top, the t-distributed stochastic neighbor embedding (t-SNE) [50] was applied to visualize the deep features and present the related representation space. The t-SNE results of each method (with the optimal hyper-parameter) on the CIFAR-10

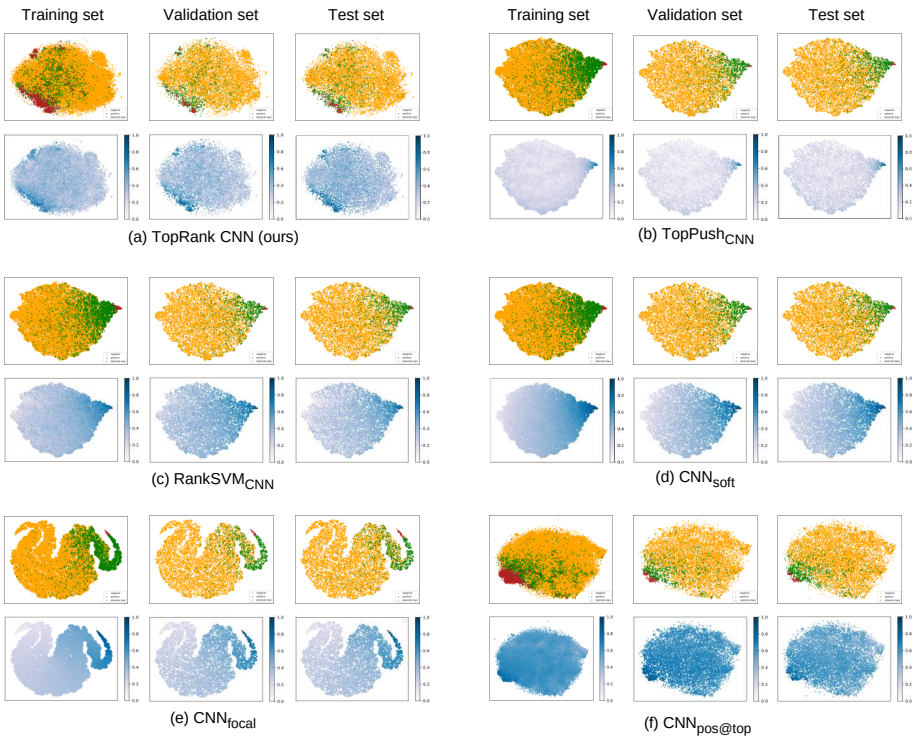


Figure 5: t-SNE visualizations of the training, validation, and test sets of the unbalanced CIFAR-10 dataset. For each method, the upper row shows positive (green and red; red denotes *the absolute top*) and negative (orange) sample distribution and the lower shows their ranking scores (darker color is higher). For the classification methods, such as $\mathrm{CNN}_{\mathrm{soft}}$, their soft-max values are used as the ranking scores. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

dataset are shown in Fig. 5. The red dots depict the absolute top samples given by each method, whereas the green dots depict positive samples that are not selected as the absolute top.

Among all the methods, TopRank CNN has more absolute top samples around the edge of the distribution owing to the allocation of larger rank scores to positive samples. The other methods also yield a reasonable separation between positive and negative samples. However, most of them (except for $\mathrm{CNN}_{\mathrm{pos@top}}$), cannot have more absolute top samples.

5. Experiment on a medical image diagnosis task

As noted in Section 1, more absolute top samples are crucial for medical image diagnosis for more reliable decisions. In this section, we apply the proposed method to a mammogram image diagnosis task to observe the practical performance of the proposed method. The experimental results show that the proposed method TopRank CNN can have a higher Pos@Top and is also very robust to this unbalanced diagnosis task.

5.1. Dataset

The DDSM mammography open-source dataset [2] was used to evaluate the performance of the proposed method in practical medical-image-based diagnosis tasks. Fig. 6 shows several samples from this dataset. The DDSM mammography dataset is a large-scale unbalanced binary mammogram image dataset that consists of negative samples (normal images) from the DDSM dataset [51] and positive samples (abnormal images) from the CBIS-DDSM dataset [52]. The sample images are preprocessed and converted to 299×299 pixels by extracting the regions-of-Interest (ROIs). It contains $55,885$ training samples, of which 14% are positive samples, and the remaining 86% are negative samples. Accordingly, the ratio of the positive and negative samples on the training set is $1:7$. The test set consists of $15,364$ images, which are constructed by $2,004$ masses or calcification images (positive) and $13,360$ normal images (negative).

5.2. Comparative methods

TopRank CNN is evaluated against four comparative methods, including CNN_{soft}, $\text{RankSVM}_{\text{CNN}}$, $\text{TopPush}_{\text{CNN}}$, and $\text{CNN}_{\text{focal}}$. In this experiment, the backbone architecture of CNN is the fully convolutional network (FCN) applied to recognize abnormalities from the DDSM mammography dataset [3] that comprised 10 convolutional layers. As proved in the case of the unbalanced CIFAR-10 dataset, the loss function of (4) with larger p approaches to the TopRank loss

[2] https://www.kaggle.com/skooch/ddsm-mammography

[3] https://www.kaggle.com/skooch/fcn-for-detecting-abnormalities-in-mammograms

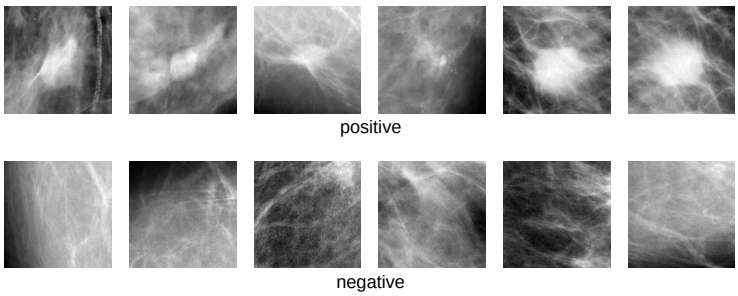


Figure 6: Examples from the DDSM Mammography dataset. The upper row shows the masses or calcification images (positive) and the bottom row shows the normal images (negative).

(5) may lead to overfitting problems. Thus, we did not conduct the comparative experiment with $\mathrm{CNN}_{\mathrm{pos@top}}$ in this part.

Five-fold cross-validation was used for model validation based on the use of the same settings of hyper-parameters as reported in the reference literature [6, 7]. According to the validation, λ was selected from $\{10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}\}$ and C was selected from $\{10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}\}$. In consideration of the numerical overflow problem about the p-norm with larger p values, in these experiments, p was selected from the set of $\{2, 4, 8, 16, 32\}$.

5.3. Results on the medical image diagnosis task

The quantitative evaluation results are shown in Table 2(a). For Pos@Top, TopRank CNN largely outperforms $\mathrm{CNN}_{\mathrm{soft}}$, $\mathrm{TopPush}_{\mathrm{CNN}}$, $\mathrm{RankSVM}_{\mathrm{CNN}}$, and $\mathrm{CNN}_{\mathrm{focal}}$, as expected. Additionally, the AUC of the TopRank CNN was almost the same as $\mathrm{RankSVM}_{\mathrm{CNN}}$ (which was designed to maximize AUC).

Fig. 7 shows the ROC of each method. As explained (Fig. 2), the leftmost vertical part of the ROC was Pos@Top. This is very important, especially for medical diagnosis, as Pos@Top presents the ratio of absolute top samples (which are very reliable positive samples). These ROC plots also support the fact that the proposed method can find more absolute top samples than the other methods.

Fig 8 shows the top-10-ranked medical image samples among the test set. Note that all of them are positive samples (according to their ground-truth

Table 2: *Pos@Top* and AUC of TopRank CNN (ours) and comparative methods on the DDSM Mammography dataset.

(a) Result on the original DDSM Mammography test set (the ratio of positive and negative samples is $1:7$).

Method	Feature	Algorithm	Loss	Hyper-param.	AUC	Pos@Top
$\text{TopPush}_{\text{CNN}}$	CNN_{soft}	TopPush	Eq.(4)	$\lambda = 0.1$	0.9317	0.1362
$\text{RankSVM}_{\text{CNN}}$	CNN_{soft}	RankSVM	Eq.(2)	$C = 10$	**0.9799**	0.1626
CNN_{soft}	end-to-end	CNN	Softmax	Epoch=100	0.9780	0.1043
$\text{CNN}_{\text{focal}}$	end-to-end	CNN	Focal	Epoch=100	0.9713	0.1290
TopRank CNN (ours)	end-to-end	CNN	Eq.(5)	$p = 32$, Epoch=30	0.9791	**0.3183**

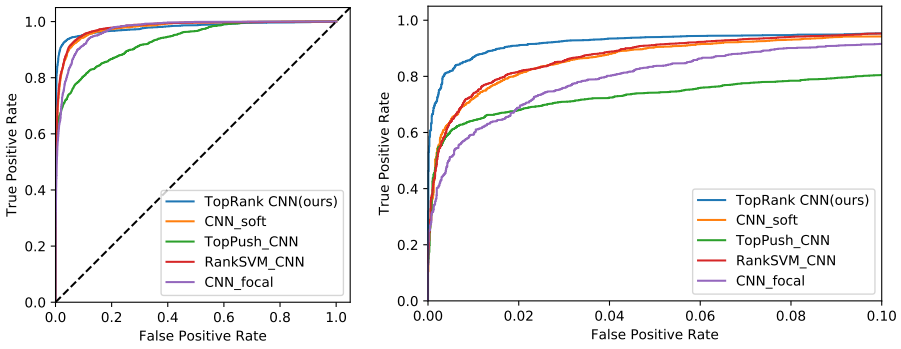
(b) Result when the extremely unbalanced DDSM Mammography training samples (whose the ratio of positive and negative samples is $1:97$) are used.

Method	Feature	Algorithm	Loss	Hyper-param.	AUC	Pos@Top
$\text{TopPush}_{\text{CNN}}$	CNN_{soft}	TopPush	Eq.(4)	$\lambda = 0.1$	0.8544	0.0115
$\text{RankSVM}_{\text{CNN}}$	CNN_{soft}	RankSVM	Eq.(2)	$C = 0.01$	0.9014	0.0050
CNN_{soft}	end-to-end	CNN	Softmax	Epoch=100	0.9319	0.0000
$\text{CNN}_{\text{focal}}$	end-to-end	CNN	Focal	Epoch=100	0.8315	0.0000
TopRank CNN (ours)	end-to-end	CNN	Eq.(5)	$p = 8$, Epoch=50	**0.9523**	**0.1023**

images) and belong to the absolute top.

Fig. 9 shows the t-SNE visualization of the sample distributions in the feature spaces of individual methods. Just like the unbalanced CIFAR-10 dataset, it is confirmed that TopRank CNN could form a larger area for absolute top samples than the other methods.

Finally, we observed the effects of p-norm relaxation on the maximization of Pos@Top. Fig. 10 shows Pos@Top for the validation set with different p values. Given that a larger p will mimic the original loss function (5) to maximize Pos@Top, Pos@Top becomes larger along with p. However, we can also observe that Pos@Top is almost saturated after $p = 16$. This fact is very favorable for our TopRank CNN because its p-norm relation loss (4) suffers from numerical overflows at larger p values.



(a) (b)

Figure 7: (a) ROC of TopRank CNN (ours) and comparative methods on the test set of the DDSM mammography dataset. (b) The leftmost vertical part of (a).

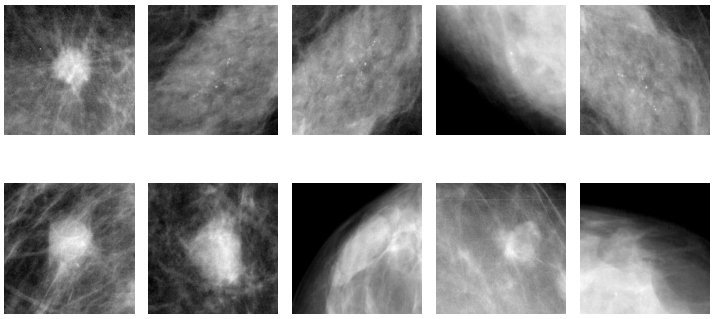


Figure 8: Top-10 ranked samples in the test set by the TopRank CNN (ours). All of these samples are abnormal (positive) images according to their ground-truth.

5.4. Results on extremely unbalanced data

The experiments described have proved the superiority of TopRank CNN for solving medical image diagnosis tasks. Specifically, the minorities (the positive samples) are not ignored but rather emphasized based on its training process. Finally, we could have larger absolute top samples. A new question arises: what will happen if the minorities are more minor? In other words, is it worthy to observe the performance of TopRank CNN on extremely unbalanced data?

For this observation, we created an extremely unbalanced dataset from the original DDSM mammography dataset. Specifically, we randomly picked 500

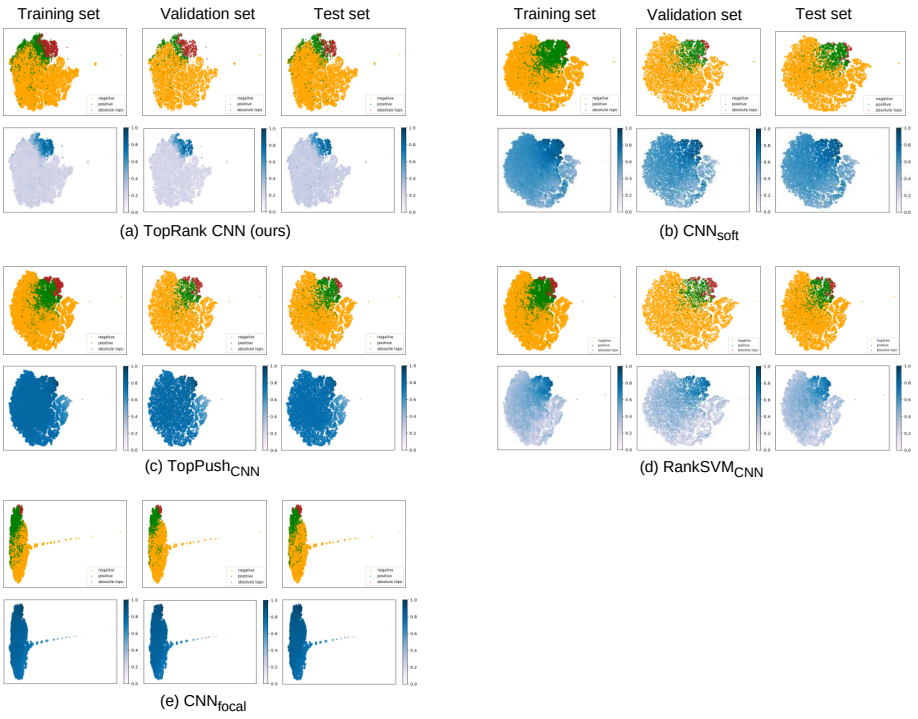


Figure 9: t-SNE visualization of the training, validation, and test sets of the DDSM Mammography dataset. For each method, the upper row shows positive (green and red; red denotes *the absolute top*) and negative (orange) sample distribution and the lower shows their ranking scores (darker color is higher).(For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

positive images and all the negative images from the training set to build the new training set. This dataset was composed of 1.02% positive samples and 98.98% negative samples. The ratio of positive and negative samples of training data was changed from $1:7$ to $1:97$. The original test set was then used to test the performance of each method. Except for the training data and class weight hyper-parameter, we applied the same experimental settings with the previous experiment, including comparative methods and evaluation metrics, as well as the CNN architecture.

Table 2(b) shows the test results of Pos@Top and AUC, and it confirms that the TopRank CNN still outperforms the other methods based on its Pos@Top

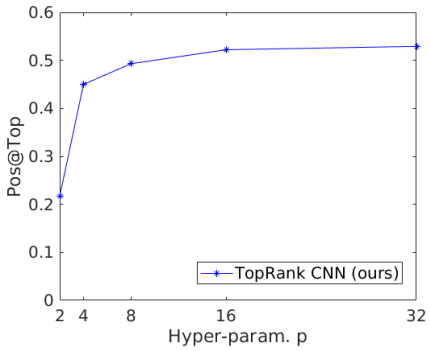
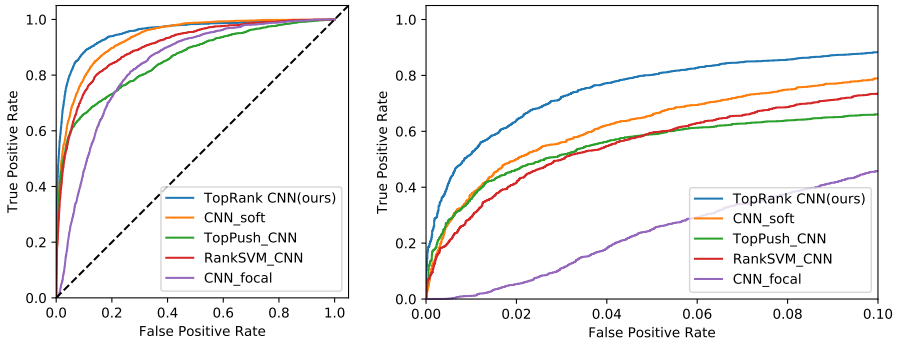


Figure 10: Pos@Top of TopRank CNN (ours) with different p on the validation set of the DDSM Mammography dataset.



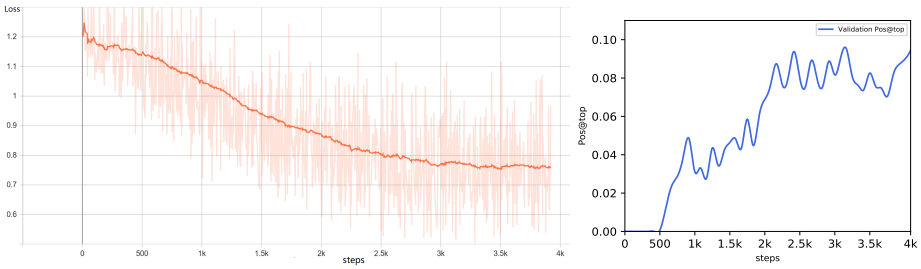
(a) (b)

Figure 11: (a) ROC of TopRank CNN (ours) and comparative methods on the test set when the extremely unbalanced DDSM mammography data is used for training. (b) The leftmost part of (a).

and AUC values. Although the Pos@Top of TopRank CNN becomes smaller than the original dataset owing to this extremely unbalanced condition, its Pos@Top is far larger than the other methods.

This superiority is also confirmed by the ROC shown in Fig. 11. The lower ROC outcomes of the comparative methods indicates their difficulties to identify the reliable, positive samples (i.e., absolute top samples).

For $\text{CNN}_{\text{focal}}$, we tried different hyperparameters for experiments, but significant improvements were not obtained. The corresponding results shown in the paper were achieved by setting the gamma and alpha to 2 and 0.25 due to the best performance of Focal loss in the literature [20]. (Note that although



(a) (b)

Figure 12: (a)The training loss curve of the proposed TopRank CNN with p=8 when the extremely unbalanced DDSM Mammography training samples (whose the ratio of positive and negative samples is 1 : 97) are used. (b) The corresponding pos@top curve on validation data at different steps.

we conducted a further hyperparameter search, the best hyperparameter values were found as alpha=0.99 and gamma=0; this means that the focal loss is reduced to the softmax loss for this dataset. AUC and pos@top for the case were 0.8823 and 0.002, respectively.)

To ensure the convergence of TopRank loss in such an extreme unbalanced situation, we presented the training loss curve and validation performance of the proposed TopRank CNN in Fig. 12. When steps reached 4k, the training loss value became stable and the pos@top on validation data also reached the peak and stabilized in a small fluctuation range.

6. Conclusion

This study proposed the *Top-rank convolutional neural network* (TopRank CNN) that maximized the absolute top samples that were very reliable positive samples. This is thought to be the first discussion of a top-rank learning approach with representation learning in an end-to-end way. Featuring a combination of top-rank learning and representation learning with p-norm relaxation, TopRank CNN maximized the absolute top samples, while it formed the appropriate feature space. As shown by the experimental results on medical image-based diagnosis tasks, TopRank CNN outperformed other top-rank

learning methods and conventional deep CNNs based on the higher ratios of absolute top samples (i.e., Pos@Top) and higher AUC values. The robustness to unbalanced tasks was also proved experimentally.

Future work will focus on the improvement of TopRank CNN and its application to multi-class recognition tasks. In addition, as we discussed in Section 3.2, the performance of TopRank CNN on the maximization of Pos@Top is often disturbed by outliers during the training. As an extension of this study, we will apply some techniques to reduce the influence of the outliers and enhance the effectiveness of the TopRank CNN.

Acknowledgments

This work was supported by JSPS KAKENHI (Grant Number JP17H06100 and JP18K18001) and China Scholarship Council (Grant Number 201806330079).

References

[1] S. Agarwal, P. Niyogi, Stability and generalization of bipartite ranking algorithms, in: Proceedings of the COLT, 2005, pp. 32–47. doi:10.1007/11503415_3.

[2] K. Uematsu, Y. Lee, On theoretically optimal ranking functions in bipartite ranking, J. Am. Stat. Assoc 112 (519) (2017) 1311–1322. doi:10.1080/01621459.2016.1215988.

[3] H. He, Y. Ma, Imbalanced Learning: Foundations, Algorithms, and Applications, 2013.

[4] S. Clémençon, N. Vayatis, Ranking the best instances, J. Mach. Learn. Res. 8 (2007) 2671–2699.

[5] S. P. Boyd, C. Cortes, M. Mohri, A. Radovanovic, Accuracy at the top, in: Proceedings of the NIPS, 2012, pp. 962–970.

[6] N. Li, R. Jin, Z. Zhou, Top rank optimization in linear time, in: Proceedings of the NIPS, 2014, pp. 1502–1510.

[7] T. Joachims, Optimizing search engines using clickthrough data, in: Proceedings of the SIGKDD, 2002, pp. 133–142. doi:10.1145/775047.775067.

[8] C. J. C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, G. N. Hullender, Learning to rank using gradient descent, in: Proceedings of the ICML, 2005, pp. 89–96. doi:10.1145/1102351.1102363.

[9] T. Liu, Learning to rank for information retrieval, Found. Trends Inf. Retr. 3 (3) (2009) 225–331. doi:10.1561/1500000016.

[10] H. Li, Learning to rank for information retrieval and natural language processing, Synthesis Lectures on Human Language Technologies 4 (1) (2011) 1–113. doi:10.2200/S00348ED1V01Y201104HLT012.

[11] S. Agarwal, T. Graepel, R. Herbrich, S. Har-Peled, D. Roth, Generalization bounds for the area under the ROC curve, J. Mach. Learn. Res. 6 (2005) 393–425.

[12] A. K. Menon, R. C. Williamson, Bipartite ranking: a risk-theoretic perspective, J. Mach. Learn. Res. 17 (2016) 1–102.

[13] J. Hanley, B. McNeil, The meaning and use of the area under a receiver operating characteristic (roc) curve, Radiology 143 (1) (1982) 29–36. doi:10.1148/radiology.143.1.7063747.

[14] C. Cortes, M. Mohri, AUC optimization vs. error rate minimization, in: Proceedings of the NIPS, 2003, pp. 313–320.

[15] R. Longadge, S. Dongre, Class imbalance problem in data mining review, arXiv preprint arXiv:1305.1707.

[16] E. Ahmed, M. J. Jones, T. K. Marks, An improved deep learning architecture for person re-identification, in: Proceedings of the CVPR, 2015, pp. 3908–3916. doi:10.1109/CVPR.2015.7299016.

[17] L. G. Hafemann, R. Sabourin, L. S. Oliveira, Learning features for offline handwritten signature verification using deep convolutional neural networks, Pattern Recognit. 70 (2017) 163–176. doi:10.1016/j.patcog.2017.05.012.

[18] M. Zhu, J. Xia, X. Jin, M. Yan, G. Cai, J. Yan, G. Ning, Class weights random forest algorithm for processing class imbalanced medical data, IEEE Access 6 (2018) 4641–4652. doi:10.1109/ACCESS.2018.2789428.

[19] J. Van Hulse, T. M. Khoshgoftaar, A. Napolitano, Experimental perspectives on learning from imbalanced data, in: Proceedings of ICML, 2007, pp. 935–942. doi:10.1145/1273496.1273614.

[20] T. Lin, P. Goyal, R. B. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the ICCV, 2017, pp. 2999–3007. doi:10.1109/ICCV.2017.324.

[21] K. Oksuz, B. C. Cam, S. Kalkan, E. Akbas, Imbalance problems in object detection: a review, IEEE Trans. Pattern Anal. Mach. Intell.doi:10.1109/TPAMI.2020.2981890.

[22] A. Rakotomamonjy, Optimizing area under roc curve with svms, in: Proceedings of ROCAI, 2004, pp. 71–80.

[23] R. Cruz, K. Fernandes, J. F. P. da Costa, M. Pérez-Ortiz, J. S. Cardoso, Combining ranking with traditional methods for ordinal class imbalance, in: Proceedings of the IWANN, 2017, pp. 538–548. doi:10.1007/978-3-319-59147-6_46.

[24] Y. Cao, J. Xu, T. Liu, H. Li, Y. Huang, H. Hon, Adapting ranking SVM to document retrieval, in: Proceedings of the SIGIR, 2006, pp. 186–193. doi:10.1145/1148170.1148205.

[25] C. Lee, C. Lin, Large-scale linear ranksvm, Neural Comput. 26 (4) (2014) 781–817. doi:10.1162/NECO_a_00571.

[26] R. Cruz, K. Fernandes, J. S. Cardoso, J. F. P. da Costa, Tackling class imbalance with ranking, in: Proceedings of the IJCNN, 2016, pp. 2182–2187. doi:10.1109/IJCNN.2016.7727469.

[27] Y. Zheng, Y. Zheng, W. Ohyama, D. Suehiro, S. Uchida, Ranksvm for offline signature verification, in: Proceedings of the ICDAR, 2019, pp. 928–933. doi:10.1109/ICDAR.2019.00153.

[28] Y. Zheng, W. Ohyama, B. K. Iwana, S. Uchida, Capturing micro deformations from pooling layers for offline signature verification, in: Proceedings of the ICDAR, 2019, pp. 1111–1116. doi:10.1109/ICDAR.2019.00180.

[29] T. Joachims, A support vector method for multivariate performance measures, in: Proceedings of the ICML, 2005, pp. 377–384. doi:10.1145/1102351.1102399.

[30] H. Narasimhan, S. Agarwal, A structural SVM based approach for optimizing partial auc, in: Proceedings of the ICML, 2013, pp. 516–524.

[31] Y. Yue, T. Finley, F. Radlinski, T. Joachims, A support vector method for optimizing average precision, in: Proceedings of the SIGIR, 2007, pp. 271–278. doi:10.1145/1277741.1277790.

[32] Q. Le, A. Smola, Direct optimization of ranking measures, arXiv preprint arXiv:0704.3359.

[33] M. Xu, Y. Li, Z. Zhou, Multi-label learning with PRO loss, in: Proceedings of the AAAI, 2013.

[34] H. Valizadegan, R. Jin, R. Zhang, J. Mao, Learning to rank by optimizing ndcg measure, in: Proceedings of the NIPS, 2009, pp. 1883–1891.

[35] C. Rudin, The p-norm push: a simple convex ranking algorithm that concentrates at the top of the list, J. Mach. Learn. Res. 10 (2009) 2233–2271. doi:10.1145/1577069.1755861.

[36] S. Agarwal, The infinite push: A new support vector ranking algorithm that directly optimizes accuracy at the absolute top of the list, in: Proceedings of the SDM, 2011, pp. 839–850. doi:10.1137/1.9781611972818.72.

[37] P. Huang, X. He, J. Gao, L. Deng, A. Acero, L. P. Heck, Learning deep structured semantic models for web search using clickthrough data, in: Proceedings of the CIKM, 2013, pp. 2333–2338. doi:10.1145/2505515.2505665.

[38] A. Severyn, A. Moschitti, Learning to rank short text pairs with convolutional deep neural networks, in: Proceedings of the SIGIR, 2015, pp. 373–382. doi:10.1145/2766462.2767738.

[39] J. Guo, Y. Fan, Q. Ai, W. B. Croft, A deep relevance matching model for ad-hoc retrieval, in: Proceedings of the CIKM, 2016, pp. 55–64. doi:10.1145/2983323.2983769.

[40] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, Y. Wu, Learning fine-grained image similarity with deep ranking, in: Proceedings of the CVPR, 2014, pp. 1386–1393. doi:10.1109/CVPR.2014.180.

[41] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proceedings of the CVPR, 2015, pp. 815–823. doi:10.1109/CVPR.2015.7298682.

[42] K. Sohn, Improved deep metric learning with multi-class n-pair loss objective, in: Proceedings of the NIPS, 2016, pp. 1857–1865.

[43] S. Chen, C. Guo, J. Lai, Deep ranking for person re-identification via joint representation learning, IEEE Trans. Image Process. 25 (5) (2016) 2353–2367. doi:10.1109/TIP.2016.2545929.

[44] Y. Shen, X. He, J. Gao, L. Deng, G. Mesnil, Learning semantic representations using convolutional neural networks for web search, in: Proceedings of the WWW, 2014, pp. 373–374. doi:10.1145/2567948.2577348.

[45] L. Pang, Y. Lan, J. Guo, J. Xu, J. Xu, X. Cheng, Deeprank: A new deep architecture for relevance ranking in information retrieval, in: Proceedings of the CIKM, 2017, pp. 257–266. doi:10.1145/3132847.3132914.

[46] D. Song, W. Liu, R. Ji, D. A. Meyer, J. R. Smith, Top rank supervised binary coding for visual search, in: Proceedings of the ICCV, 2015, pp. 1922–1930. doi:10.1109/ICCV.2015.223.

[47] Y. Geng, R.-Z. Liang, W. Li, J. Wang, G. Liang, C. Xu, J.-Y. Wang, Learning convolutional neural network to maximize pos@ top performance measure (2016) 589–594.

[48] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, R. Shah, Signature verification using a "Siamese" time delay neural network, Int. J. Pattern Recognit. Artif. Intell. 7 (4) (1993) 669–688. doi:10.1142/S0218001493000339.

[49] A. Krizhevsky, V. Nair, G. Hinton, Cifar-10 (canadian institute for advanced research), URL http://www. cs. toronto. edu/kriz/cifar. html 8.

[50] L. v. d. Maaten, G. Hinton, Visualizing data using t-sne, J. Mach. Learn. Res. 9 (Nov) (2008) 2579–2605.

[51] K. Bowyer, D. Kopans, W. Kegelmeyer, R. Moore, M. Sallam, K. Chang, K. Woods, The digital database for screening mammography, in: Proceedings of the IWDM, 1996, p. 27. doi:10.1007/978-94-011-5318-8_75.

[52] R. S. Lee, F. Gimenez, A. Hoogi, K. K. Miyake, M. Gorovoy, D. L. Rubin, A curated mammography data set for use in computer-aided detection and diagnosis research, Sci. Data 4 (2017) 170177. doi:10.1038/sdata.2017.177.