Does one rotten apple spoil the whole barrel?

Veronika Cheplygina, David M.J. Tax, Marco Loog
Pattern Recognition Laboratory, Delft University of Technology, The Netherlands
{*v.cheplygina, d.m.j.tax, m.loog*}*@tudelft.nl*

Abstract

Multiple Instance Learning (MIL) is concerned with learning from sets (bags) of objects (instances), where the individual instance labels are ambiguous. In MIL it is often assumed that positive bags contain at least one instance from a so-called concept in instance space. However, there are many MIL problems that do not fit this formulation well, and hence cause traditional MIL algorithms, which focus on the concept, to perform poorly. In this work we show such types of problems and the methods appropriate to deal with either situation. Furthermore, we show that an approach that learns directly from dissimilarities between bags can be adapted to deal with either problem.

1. Introduction

Multiple-instance learning (MIL) [2] extends traditional supervised learning methods in order to learn from objects that are described by a set (*bag*) of feature vectors (*instances*), rather than a single feature vector only. MIL problems are often considered to be two-class problems, i.e., a bag of instances can belong either to the positive or the negative class. During training, the bag labels are available, but the labels of the individual instances are not defined. However, we can make several assumptions about how the ambiguous instance labels relate to the bag labels.

It is traditionally assumed that positive bags contain one or more positive instances from a so-called *concept*, whereas negative bags contain only negative instances [2, 5]. There are two assumptions here: that the concept is a region of the instance space, and that a positive bag contains at least one instance from this region. Many MIL approaches try to model such a concept by identifying the "most positive" instances in bags, and classify new bags as positive if they appear to have instances within this concept [2, 5]. It has occasionally been pointed out that MIL problems do not always adhere to these assumptions [11]. Here, the fruit disease classification problem is described. When classifying batches of apples as good (negative) or infected (positive), just one rotten apple in a batch does not necessarily mean the whole batch is infected. Or, when classifying emails as normal (negative) or spam (positive), one presence of a word such as "offer" or "bonus" might not necessarily mean the email is spam, whereas a high occurence of several of such terms might.

Dissimilarity-based approaches [6] provide new opportunities to deal with MIL problems. It is possible to define (dis)similarity measures between bags, and use these dissimilarities as features for supervised classifiers. This provides a possibility to compare positive and negative bags in a direct way rather than relying on locating a concept. Therefore, such methods might be particularly suitable for MIL problems without the classical definition of a concept. Many ways of defining dissimilarities between bags are available. We show two illustrative methods and emphasize their advantages and disadvantages for different types of MIL problems.

2. Preliminaries

In Multiple Instance Learning, an object is represented by a bag $B_i = \{\mathbf{x}_{ik} | k = 1, ..., n_i\} \subset \mathbb{R}^d$ of n_i instances. The training set $T = \{(B_i, y_i) | i = 1, ...N\}$ consists of positive ($y_i = +1$) and negative ($y_i = -1$) bags. The two traditional assumptions for MIL are as follows:

1. There is a concept $C \subset \mathbb{R}^d$, instances inside this concept are positive $\forall \mathbf{x}_{ik} \in C, y_{ik} = 1$.

2. A positive bag has at least 1 positive instance, whereas a negative bag has only negative instances: if $y_i = +1$ then $\exists x_{ik} \in B_i : y_{ik} = +1$, if $y_i = -1$ then $\forall \mathbf{x}_{ik} \in B_i y_{ik} = -1$.

The task is to find a classifier f_B. Traditionally this is achieved by finding an instance classifier f_I and by

obtaining the bag label using $\hat{y_i} = \max_j\{\hat{y_{ij}}\}$. An alternative is to combine the instance posterior probabilities by averaging, product rule, voting and so forth[4].

Earlier MIL algorithms such as [2, 5] explicitly use both assumptions. In order to classify bags, these algorithms model the concept, use this model to classify individual instances, and apply assumption 2 to obtain the bag labels. In other words, in such a formulation the "most positive" instance is responsible for the bag label. We refer to such methods as *instance-based*. Although MILES[1] does not learn on instance level, it involves selecting more informative instances, and therefore has some similarities with the traditional methods.

An alternative approach is to learn from the bags directly. In [10], several variants of the Hausdorff distance are used to define distances between bags. The label of a bag is then decided by the labels of its neighbors, and their neighbors. In [3], information is extracted from bags and SVMs are used on kernels defined on this information. For instance, the minimax kernel represents a bag by the minimum and maximum instance values of all the features. The information extracted from a bag is therefore dependent on more than 1 instance in a bag. A more general approach of using bag dissimilarities as features for a supervised classifier is explored in [9, 7]. Note that these methods do not explicitly rely on assumptions 1 and 2. We refer to these methods as *bag-based*.

3. Dissimilarities for MIL

A bag is often represented by a set of feature vectors in the instance space. However, we can also represent a bag by its dissimilarities to bags in the training set T, i.e. by a vector $\mathbf{d}(B_i, T) = [d(B_i, B_1), ...d(B_i, B_N)]$. In the dissimilarity space, each bag is represented by a single feature vector and the MIL problem can be viewed as a standard supervised learning problem.

There are several ways of defining the bag dissimilarity measure $d(B_i, B_j)$. Here we focus on defining $d(B_i, B_j)$ through the pairwise instance dissimilarities $D = [d(\mathbf{x}_{ik}, \mathbf{x}_{jl})]_{N_i \times N_j}$. We use the Euclidean distance: $d(\mathbf{x}_{ik}, \mathbf{x}_{jl}) = \| \mathbf{x}_{ik} - \mathbf{x}_{jl} \|_2$ for the instance dissimilarity, but other choices are also possible.

Two intuitive choices proposed in [9] are the overall minimum distance or *minmin*, and the average minimum distance or *meanmin*, shown in Equations 1 and 2, respectively. These dissimilarities help us to draw parallels between our method and other MIL approaches, and to show in which type of problems one of these methods may be preferable.

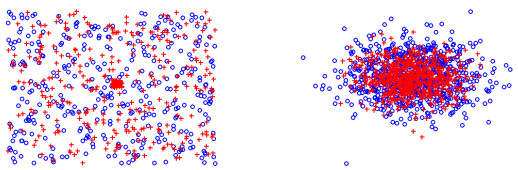

Figure 1. Artificial datasets: well-defined concept (left), overlapping distributions (right). + and ∘ are instances from positive and negative bags.

$$d_{minmin}(B_i, B_j) = \min_{k,l} d(\mathbf{x}_{ik}, \mathbf{x}_{jl}) \quad (1)$$

$$d_{meanmin}(B_i, B_j) = \frac{d(B_i, B_j) + d(B_j, B_i)}{2}$$

$$\text{where } d(B_i, B_j) = \frac{1}{n_i} \sum_{k=1}^{n_i} \min_{l} d(\mathbf{x}_{ik}, \mathbf{x}_{jl}) \quad (2)$$

We expect *minmin* and *meanmin* to do well in different situations. In problems with a well-defined, dense concept and a less dense background, such as in Figure 1 (left), *minmin* should perform well because concept instances will be much closer to each other than background instances and will always be "selected" by *minmin* to represent two positive bags. Therefore, the dissimilarities between a pair of positive bags will be lower than dissimilarities between other pairs. However, if the concept is not so dense, or is not present at all, *minmin* may fail because the dissimilarities between all pairs of bags may become very close to each other.

On the other hand, *meanmin* should be less dependent on the presence of a dense concept. As long as the positive and negative bags have different distributions of instances, as in Figure 1 (right), *meanmin* should produce different dissimilarity values for different pairs of bags. However, this becomes more challenging as the differences in distributions become more and more subtle. For instance, in a problem with a concept, if the proportion of background instances to concept instances in positive bags is very high, the concept instances might not be able to contribute enough to the overall dissimilarity to ensure that there is a separation of positive and negative bags in the dissimilarity space.

4. Experimental setup

We illustrate the differences of the instance-based and bag-based methods using two artificial datasets,

shown in Figure 1. In the *Concept-m* dataset [5], positive bags consist of a single instance inside a concept, as well as instances from a background uniform distribution, whereas the negative bags only contain background instances. The suffix m indicates the average number of background instances per bag. In the *Distribution-*α dataset, instances of positive and negative bags originate from two normal distributions $\mathcal{N}(0, \alpha)$, $\alpha \in \{0.5, 0.75\}$ and $\mathcal{N}(0, 1)$ respectively. A higher value for α corresponds to a larger overlap between the two distributions.

The *Musk* datasets[2] are examples of problems adhering to both assumptions. The bags in the data represent molecules, the instances describe the surfaces of several possible shapes of these molecules. Some molecules have certain shapes, which can make them smell musky, other molecules do not have these shapes and thus do not smell musky. Presumably the "smelly" shapes are close to each other in the space of molecule shapes, thus forming a concept.

A dataset outside the traditional definition is the *Newsgroups* classification problem [12]. A positive bag is a collection of posts, where 3% of the posts are from the actual topic of the newsgroup, and 97% are from the other topics. Each post is represented by the top word frequency features. Although assumption 2 holds, the presence of a concept here is debatable. In particular, the use of word frequency features does not guarantee a well-defined concept C: posts on the same topic do not necessarily have to use the same words, and posts on different topics may use a lot of the same words, thus there is a large overlap in the instance space.

We create the dissimilarity representations using the *minmin* and *meanmin* dissimilarities, and use Parzen and k-nearest neighbor classifiers (k-NN) on these dissimilarities. Note that k-NN in the dissimilarity space is not the same as using the dissimilarities as distances in a nearest neighbor sense directly.

Furthermore, we use Diverse Density (instance-based), MILES (bag-based with instance selection), the Minimax kernel and Citation-kNN (both bag-based) classifiers to emphasize the differences of such approaches in concept and non-concept type problems. The purpose here is mainly to demonstrate the difference of instance-based and bag-based methods, not to claim that a dissimilarity approach achieves the highest performance. The implementations of all classifiers are from [8].

5 Results and Discussion

The results are presented in Table 1. For the *Concept-*m datasets, *minmin* is able to capture the same information as the instance-based counterparts, resulting in good performance. The performance deteriorates asa m increases because the densities of concept and background grow closer. *Minmin* also performs well for *Musk*, which suggests that the concept is indeed denser than the background. For the *Distribution-*α problems, *minmin* is not a good choice. As the variances of the distributions grow closer, the minimum instance distances become more similar between the two classes. *Minmin* also fails in the *Newsgroups* datasets. Because only the most frequent words are used, positive and negative bags often contain identical instances, thus often causing the dissimilarity to be equal to 0.

Meanmin is less suitable for the *Concept* data. It starts off better than random because it is averaging over just a few instances and the concept is able to influence the overall dissimilarities. However, the increasing number of background instances quickly decreases this influence. *Meanmin* is clearly more robust in the *Distribution* datasets. As α increases and the minimum instance distances between classes become more similar, averaging over several distances helps to prevent degradation of performance. Surprisingly, *meanmin* performs well on the Musk data. This suggests either that the concept is much denser than the background (as in *Concept-7*), or that the background instances in positive and negative bags are from different distributions, as in the *Distribution* datasets. *Meanmin* also performs well on the Newsgroups data, suggesting that on average, different classes contain different distributions of words. Similar results can be observed for Minimax (we believe the use of linear SVM contributes to the success of this method) and Citation-k-NN.

Notice that Diverse Density and MILES follow similar trends as the *minmin* dissimilarity, whereas *meanmin* produces results similar to those of Minimax and Citation-k-NN. This shows that by changing one parameter (minimum or average) in a dissimilarity-based approach, we are able to mimic instance-based and bag-based MIL methods to some extent. These dissimilarities are extreme cases of considering just one instance per bag and comparing bags at a local level, or considering all instance per bag for a more global comparison. This provides possibilities for investigating the effects of a more flexible approach, where several (but not all) instances are taken into account.

6. Conclusions

We have emphasized that the class of MIL problems is larger than the traditional problems where the presence of a concept is assumed, and that in some problems, the complete distribution of instances may

Classifier→ Data ↓	minmin+ k-NN	minmin+ Parzen	meanmin+ k-NN	meanmin+ Parzen	Diverse Density	MILES	Minimax+ linear SVM	Citation- k-NN
concept 7	97.6 (0.7)	97.8 (0.5)	68.0 (3.1)	60.0 (2.6)	94.6 (1.9)	**99.9 (0.1)**	50.0 (2.5)	53.4 (2.4)
concept 15	90.4 (1.2)	**97.9 (0.7)**	51.6 (2.6)	52.8 (2.4)	83.9 (3.0)	**98.1 (0.6)**	40.9 (3.0)	55.0 (2.4)
concept 30	71.4 (2.1)	75.8 (2.5)	50.4 (2.3)	56.2 (2.4)	76.3 (3.7)	**96.8 (0.7)**	55.9 (2.3)	50.7 (2.2)
musk1	93.2 (1.3)	**94.7 (1.3)**	92.6 (1.4)	90.6 (1.8)	89.4 (1.3)	92.8 (1.4)	91.3 (1.7)	**91.6 (1.6)**
musk2	89.7 (1.4)	92.3 (1.3)	90.1 (1.8)	91.9 (1.3)	**93.2 (0.0)**	**95.3 (1.5)**	88.6 (1.7)	86.6 (1.8)
distr 0.5	77.6 (1.8)	83.5 (1.6)	**100.0 (0.0)**	**100.0 (0.0)**	88.9 (2.5)	97.4 (0.8)	**100.0 (0.0)**	**99.9 (0.1)**
distr 0.75	68.7 (2.1)	70.6 (2.3)	**91.2 (1.3)**	**91.0 (1.3)**	65.6 (2.5)	52.8 (1.6)	**92.6 (1.2)**	83.4 (1.9)
alt.ath	50.0 (0.0)	49.8 (0.6)	**87.8 (1.5)**	**88.6 (1.5)**	52.2 (2.4)	47.1 (4.5)	**87.8 (1.4)**	71.0 (1.8)
rec.mot	50.0 (0.0)	50.0 (0.0)	83.7 (1.9)	86.2 (1.8)	46.4 (2.9)	44.7 (4.8)	**91.0 (1.4)**	80.2 (1.9)
pol.mid	47.2 (1.2)	50.2 (1.3)	80.8 (2.0)	84.4 (1.8)	40.2 (2.5)	54.1 (1.8)	**90.4 (1.2)**	77.2 (1.7)

Table 1. AUC and standard error (x100), 5x10-fold cross-validation. Bold = best (or not significantly worse) result per dataset. The trend is that minmin and instance-based methods do well in "Concept" situations, and meanmin and bag-based methods do well in "Distribution" situations.

be more important. In such problems, methods that (incorrectly) assume the presence of a concept may fail, whereas methods that do not make this assumption and consider bags as a whole, are likely to be more successful. Such approaches include kernel, distance or dissimilarity-based methods. The latter approach works by defining dissimilarities between bags, and using standard supervised classifiers in the resulting dissimilarity space.

This paper shows how two intuitive bag dissimilarities are able to cover both the concept and distribution MIL situations. The overall minimum distance considers only a single instance per bag and is suitable for problems with a well-defined concept, whereas the mean minimum distance considers all instances in a bag and is more suitable for problems where positive and negative bags have different instance distributions. These dissimilarities are extreme cases in terms of which instances play a role in defining a dissimilarity. However, there are possibilities to define more flexible dissimilarities somewhere in between these of two extremes, therefore allowing us to trade-off more local or more global properties of bags. It remains an open question how to define such a dissimilarity and how to decide whether it is suitable for a particular type of MIL problem.

References

[1] Y. Chen, J. Bi, and J. Wang. Miles: Multiple-instance learning via embedded instance selection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(12):1931–1947, 2006.

[2] T. Dietterich, R. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1-2):31–71, 1997.

[3] T. Gärtner, P. Flach, A. Kowalczyk, and A. Smola. Multi-instance kernels. In *Proc. of the 19th Int. Conf. on Machine Learning*, pages 179–186, 2002.

[4] M. Loog and B. Van Ginneken. Static posterior probability fusion for signal detection: applications in the detection of interstitial diseases in chest radiographs. In *Proc. of the 17th Int. Conf. on Pattern Recognition*, volume 1, pages 644–647. IEEE, 2004.

[5] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. In *Advances in neural information processing systems*, pages 570–576. Morgan Kaufmann Publishers, 1998.

[6] E. Pekalska and R. Duin. *The dissimilarity representation for pattern recognition: foundations and applications*, volume 64. World Scientific Pub Co Inc, 2005.

[7] L. Sørensen, M. Loog, D. Tax, W. Lee, M. de Bruijne, and R. Duin. Dissimilarity-based multiple instance learning. *Structural, Syntactic, and Statistical Pattern Recognition*, pages 129–138, 2010.

[8] D. Tax. MIL, a Matlab toolbox for multiple instance learning, May 2011. version 0.7.9.

[9] D. Tax, M. Loog, R. Duin, V. Cheplygina, and W. Lee. Bag dissimilarities for multiple instance learning. *Similarity-Based Pattern Recognition*, pages 222–234, 2011.

[10] J. Wang and J. Zucker. Solving multiple-instance problem: A lazy learning approach. 2000.

[11] X. Xu. *Statistical learning in multiple instance problems*. PhD thesis, the University of Waikato, 2003.

[12] Z. Zhou, Y. Sun, and Y. Li. Multi-instance learning by treating instances as non-iid samples. In *Proc. of the 26th Int. Conf. on Machine Learning*, pages 1249–1256. ACM, 2009.