Combining Instance Information to Classify Bags

Veronika Cheplygina, David M.J. Tax, Marco Loog

Pattern Recognition Laboratory, Delft University of Technology
{v.cheplygina, d.m.j.tax, m.loog}@tudelft.nl

Abstract. Multiple Instance Learning is concerned with learning from sets (bags) of feature vectors (instances), where the bags are labeled, but the instances are not. One of the ways to classify bags is using a (dis)similarity space, where each bag is represented by its dissimilarities to certain prototypes, such as bags or instances from the training set. The instance-based representation preserves the most information, but is very high-dimensional, whereas the bag-based representation has lower dimensionality, but risks throwing away important information. We show a connection between these representations and propose an alternative representation based on combining classifiers, which can potentially combine the advantages of the other methods. The performances of the ensemble classifiers are disappointing, but require further investigation. The bag-based representation preserves sufficient information to classify bags correctly and produces the best results on several datasets.

1 Introduction

Multiple-instance learning (MIL) [7] 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. For example, instead of representing an image or a document by a single feature vector, we could represent each segment or paragraph by its own feature vector. This is a more flexible representation, that can potentially preserve more information than if we were to compress all segments or paragraphs into a single feature vector.

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. The bag labels are available, but the labels of the individual instances are not defined. The standard assumption here is that a bag is positive if and only if at least one instance inside the bag is positive. For example, an image labeled as "cat" would have a cat in at least one of its segments, whereas images without this label would not portray any cats at all. In this setting, it is possible to say that only one instance (the segment containing the cat) is informative.

It has been argued that there are more general kinds of MIL problems where the assumption above does not apply [21, 5]. For example, for an image of the category "beach", it would be difficult to say which part of the image is informative. We would need to identify several objects (such as water and sand) to say

that it is a beach, so at least a few instances in a positive bag must be informative. This reasoning can be extended even further to consider cases where simply the presence of particular objects is not enough: consider how much of an image has to be covered by trees for you to call it a forest. Here, a certain fraction of instances is required for the positive class label[21], and therefore most, or even all instances can be informative.

One of the ways to classify bags in MIL problems is by representing the bags in a similarity or dissimilarity space [17]: each bag is then represented by its dissimilarities to certain prototypes. In our work[20], these prototypes are (a subset of) bags from the training set. Because a single dissimilarity is defined between two bags, information provided by the more informative instances in the bags might be overlooked. In MILES [5], an alternative representation using all the instances from the training set as the prototypes is used. A 1-norm SVM is then used to automatically select the most informative dissimilarities (and therefore instances). More investigation into the instance-based representation with other base classifiers has been done in [10].

A challenge in both settings is how to define the (dis)similarity measure between a bag and a prototype. In MILES, the similarity of a bag and a prototype instance is determined by the minimum distance between the bag's instances and the prototype instance. In our work[20, 6], we define the dissimilarity of two bags as the combination (such as minimum, average or maximum) of minimum instance distances between these bags.

The way the information from different instances is combined links the instance-based and bag-based dissimilarity representations. In the former case, dissimilarities are concatenated, thus extending the dissimilarity representation, whereas in the latter they are combined into a single number by an operation such as averaging[16]. We also investigate a third alternative, i.e., combining classifiers trained on different subsets of dissimilarities. Comparing these representations can help us gain more insight into the informativeness of bags or instances as prototypes, and thus improve performances on real-life MIL problems.

2 Dissimilarity Representations

2.1 In Multiple Instance Learning

In Multiple Instance Learning, an object is represented by a bag $B_i = \{x_{ik} | k = 1, ..., n_i\} \subset \mathbb{R}^d$ of n_i feature vectors or 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 traditional assumption for MIL is that there are instance labels y_{ik} which relate to the bag labels as follows: a bag is positive if and only if it contains at least one positive, or *concept* instance[7]. In this case, it might be worthwhile to search for only these informative instances. Alternative formulations, where a fraction or even all instances are considered informative, have also been proposed [9].

We can represent an object, and therefore also a MIL bag B_i, by its dissimilarities to prototype objects in a representation set R[17]. In our work, R

is taken to be a subset of size M of the training set T of size N ($M \leq N$). Each bag is represented as $\mathbf{d}(B_i, T) = [d(B_i, B_1), ...d(B_i, B_M)]$: a vector of M dissimilarities. Therefore, each bag is represented by a single feature vector and the MIL problem can be viewed as a standard supervised learning problem.

MILES [5] considers a different definition of prototypes, using all the instances in the training set. The motivation is that, with just a few concept instances per bag, it is better to consider just these informative instances rather than the bag as a whole. MILES is originally a similarity-based approach, but in its dissimilarity-based counterpart, each bag would be represented as $\mathbf{d}(B_i, T) = [d(B_i, x_{1,1}), d(B_i, x_{1,2}), ..., d(B_i, x_{1,n_1}), ...d(B_i, x_{M,n_M})]$.

2.2 In Combining

When several dissimilarity representations for the same data are available, it can be an advantage to combine these sources of information. Assume that we are given L dissimilarity representations $D^1, D^2, \cdots, D^L$. In [16], three main ways of combining such representations are outlined:

- Concatenating the representations: $D^{ext} = [D^1 D^2 \cdots D^L]$.
- Averaging the representations: $D^{sum} = \sum_{i=1}^{L} D^i$.
- Training a base classifier on each D^i and combining the L outputs using a fixed rule (such as averaging) or a trained combiner [13, 8].

3 Approach

In previous work[20, 6], we have focused on defining $d(B_i, B_j)$ through the pairwise instance dissimilarities $[d(\mathbf{x}_{ik}, \mathbf{x}_{jl})]_{n_i \times n_j}$. We use the squared Euclidean distance for the instance dissimilarity, but other choices are also possible. In all the dissimilarities considered, the first step is to find, for each instance in B_i, the distance to its closest instance in B_j. Using these minimum instance distances, we can define many bag dissimilarities, for instance, by averaging these minimum distances. Assume that we are only given one prototype B_j. With the bag dissimilarity, the bag representation of B_i using prototype B_j would be:

$$D^{bag}_{B_j}(B_i) = \frac{1}{n_i} \sum_{k=1}^{n_i} \min_l d(\mathbf{x}_{ik}, \mathbf{x}_{jl}) \tag{1}$$

In MILES, the similarity between a bag and a prototype instance is defined as the maximum similarity between the bag's instances and the prototype instance: $s(B_i, x) = \max_k \exp\left(-\frac{d(x_{ik}, x)}{\sigma^2}\right)$. In terms of distances this corresponds to the minimum instance distance between the bag and the prototype. Therefore, the instance representation of B_i using the instances of B_j would be:

$$D^{inst}_{B_j}(B_i) = [\min_l d(\mathbf{x}_{i1}, \mathbf{x}_{jl}), \min_l d(\mathbf{x}_{i2}, \mathbf{x}_{jl}), \cdots, \min_l d(\mathbf{x}_{in_i}, \mathbf{x}_{jl})] \tag{2}$$

It is not difficult to now see that $D^{bag}_{B_j}(B_i) = \frac{1}{n_i}\sum_{k=1}^{n_i} D^{inst}_{B_j}(B_i)$. Another way to see this is that with D^{inst}, we can potentially give every prototype instance a different weight, whereas in D^{bag}, all instances from the same bag get the same weight.

Note that averaging as in (1) is not the only way to condense several dissimilarities into a single value: for instance, minimum or maximum operations could also achieve the same goal. However, these would essentially select a single instance per bag, rather than combining the information from all instances, as in (2). Therefore, we chose averaging as a combiner.

Previous results[20, 5] suggest that both the bag-based and instance-based representations are (at least partly) informative: there are at least some prototypes (bags or instances) that distinguish between positive and negative bags in the dissimilarity space. We believe that comparing D^{bag} and D^{inst} directly, we can gain more insight into the structure of Multiple Instance Learning problems: how many instances are informative and what is a good (bag or instance) prototype.

Furthermore, we introduce two other representations that can help us in this understanding. In the "bag set" representation D^{BS}, a separate classifier is built on the instances of each prototype, to form M classifiers in total. In the "random set" representation D^{RS}, random sets of instances are used to build M separate classifiers. Each set of classifiers (built on bag sets or on random sets) forms an ensemble, where the individual classifier decisions are combined.

A diagram clarifying all the representations is shown in Fig.1. In terms of the initial dissimilarity matrix, D^{inst}, D^{BS} and D^{RS} are identical, but D^{inst} is used as a single input to a single classifier, whereas D^{BS} and D^{RS} have several feature subsets and classifiers associated with them. In fact, D^{RS} is just D^{inst} used together with the random subspace method [11].

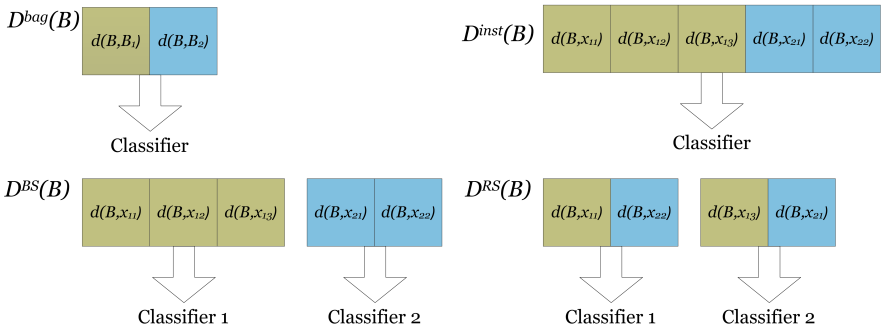

Fig. 1. Different ways for constructing dissimilarity representations of bag B using two prototype bags (green with 3 instances and blue with 2 instances). D^{bag} consists of just two dissimilarities (one for each bag), whereas D^{inst} consists of dissimilarities to all 5 instances. In D^{BS}, a separate classifier is built on each prototype's instance dissimilarities. In D^{RS}, classifiers are built on random selections of all available instances.

These representations are also interesting in terms of speed and information trade-off. We assume that the data is available offline, so that all dissimilarity matrices can be computed beforehand. D^{inst} contains all instance information, but is very high-dimensional, which can severely slow down and/or deteriorate the performance of many classifiers. D^{bag} might lose some information, but is a more compact representation, reducing training time and the possibility of overfitting. The ensembles D^{RS} and D^{BS} have access to all the information, although the information is now split up into subspaces. Although several classifiers have to be trained, each classifier can be very fast due to the reduced dimensionality, and the greater choice of classifiers that could be applied.

Alternatively, dimensionality reduction or rather, prototype selection techniques could be applied to D^{bag} or D^{inst} directly. This adds several more variables to the problem under investigation: which method for selection is used, and how many prototypes are selected. We do not pursue this line of investigation further, but we refer the reader to [18] for an overview of possible techniques.

4 Experiments

4.1 Artificial Data

Fig.2 shows two artificial datasets that help to gain some more understanding about the different representations. The first dataset originates from [15] and shows a classical concept in the middle of the plot. We call this the "Concept" dataset. A positive bag here consists of one such concept instance, the other $n_i - 1$ instances are from the background distribution, whereas negative bags have n_i instances from the background. In the second datasets, instances of positive and negative bags are generated by two Gaussians with the same mean, but different variance. We call this the "Distribution" dataset.

The Concept dataset has N bags with 25 instances each. Due to the dense concept, distances of the concept instances are informative: they are lower for positive bags, than for negative bags. In this case, a sparse classifier used on the $N \times 25N$ matrix D^{inst} should be able to find these informative distances. Averaging over the distances as in the $N \times N$ matrix D^{bag}, however, would dilute this important information. Indeed, from the learning curve we can see that D^{bag} performs very poorly in this case.

The Distribution dataset also has bags with 25 instances each. Here, the bag as a whole is a more discriminative source of information than a particular instance, because the distributions overlap. D^{inst} and D^{bag} would both contain the necessary information to classify the bag correctly, so the extra flexibility of D^{inst} would only result in more computation, not better classifiers. The learning curve also demonstrates that D^{bag} provides enough information for good performance.

4.2 Real-life MIL Data

We test all representations on several MIL datasets. Because of the number of different experiments and the running times using the instance-based represen-

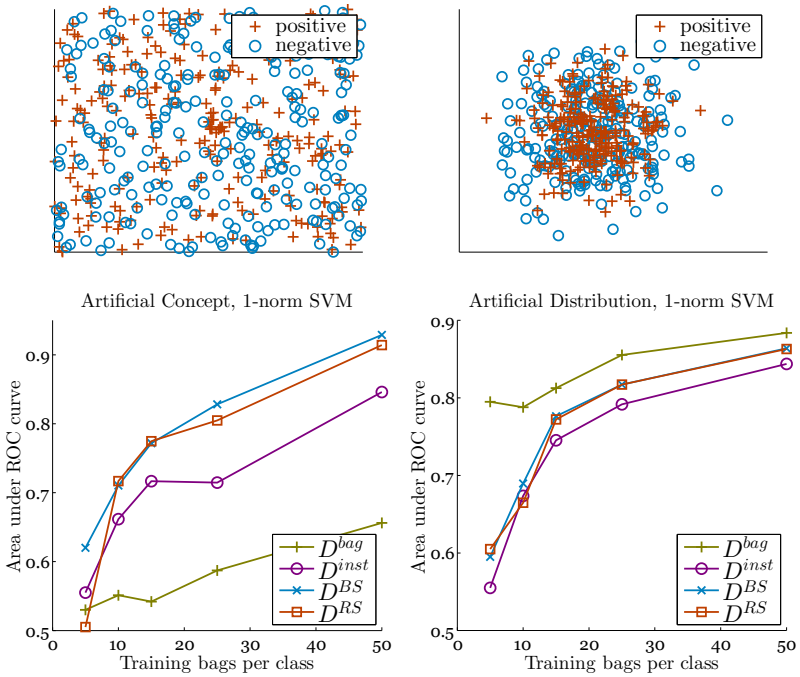

Fig. 2. Artificial datasets and corresponding learning curves. In the datasets, + and o are instances of positive and negative bags respectively.

tation D^{inst} and, we limit ourselves to a few MIL datasets with a reasonable total number of instances. A list is shown in Table 1.

The Musk datasets[7] are molecule activity prediction problems, where bags are molecules and instances are different conformers (thus with different activity) of these molecules. Fox, Tiger and Elephant are image datasets, where the bags are images and instances are segments (of which at least some segments contain foxes, tigers or elephants). These datasets are strongly expected to have a concept, and methods that explicitly search for concept instances, have been quite successful.

In Newsgroups[23] and Web Recommendation [22], both text categorization datasets, the situation might be different. In Newsgroups, a bag is a collection of posts where a post is represented by counts of frequently-occurring words. At the first glance, it seems that this is a typical Concept-type dataset: a positive bag for the category "politics" contains 3% of posts about politics, whereas negative bags contain only posts about other topics. What is different here, is that posts about politics may have nothing in common and thus be very far apart in the feature space, unlike the concept instances in the artificial Concept dataset.

In the bird song datasets [3], a bag is a audio fragment consisting of bird songs of different species. Whenever a particular species is heard in the fragment,

Dataset	+bags	-bags	total	min	mean	max
Musk 1 [7]	47	45	476	2	5	40
Musk 2 [7]	39	63	6598	1	65	1044
Fox [1]	100	100	1302	2	7	13
Tiger[1]	100	100	1220	1	6	13
Elephant [1]	100	100	1391	2	7	13
Alt.atheism [23]	50	50	5443	22	54	76
Rec.motorcycles [23]	50	50	4730	22	47	73
Politics.mideast [23]	50	50	3376	15	34	55
Web recommendation 1 [22]	21	92	2212	4	30	131
Web recommendation 4 [22]	88	25	2291	4	31	200
Web recommendation 7 [22]	54	59	2400	4	32	200
Brown Creeper [3]	197	351	10232	2	19	43
Winter Wren [3]	109	439	10232	2	19	43
Pacific slope Flycatcher [3]	165	383	10232	2	19	43

Table 1. MIL Datasets. Number of positive and negative bags as well as the total, minimum, average and maximum number of instances per bag are specified.

the bag is positive for that category. It could be expected that birds of the same species have similar songs, therefore there should be different concepts for different bird species. It is also possible that some species are heard together more often.[1] In this case, instances which are negative for one species, could still be helpful in classifying fragments as containing that species or not.

We want to compare different data representations using the same base classifier, therefore, this classifier should be applicable to both large (D^{inst}) and small (D^{BS}, as some bags may contain just 2 or 3 instances) dimensionalities. We use the 1-norm SVM (or Liknon classifier [2]) and the Winnow classifier [14] as classifiers which are able to select a few informative dissimilarities. Furthermore, we use the logistic classifier and the support vector classifier LIBSVM [4] with a linear kernel to compare the results when no such explicit selection is taking place.

Each dataset and classifier combination is tested using the four representations D^{bag}, D^{inst}, D^{BS} and D^{RS}. For D^{RS} we let the number of classifiers is equal to the number of bags (just as in D^{BS}), the number of instances for each subspace is set to the average number of instances per bag. Both ensembles are combined by averaging the posterior probabilities of the individual classifiers. These settings are chosen as reasonable default settings for a fair comparison. We use the area under the receiver-operating-characteristic (AUC) as the evaluation measure, because this is found to be more discrimnative between classifiers[12] and more suitable for MIL problems[19]. Note that many other MIL papers use the accuracy as the evaluation measure and the results cannot be compared directly.

[1] We have verified this, and this is indeed true for some species, e.g. the labels of Winter Wren and Pacific-slope Flycatcher have a correlation of 0.63

5 Results and Discussion

	Dataset	D^{bag}	D^{inst}	D^{BS}	D^{RS}
Winnow	Musk1	**89.6 (1.6)**	**90.2 (1.4)**	83.6 (1.8)	83.1 (1.9)
	Musk2	83.6 (1.9)	84.0 (2.4)	85.6 (2.2)	**88.4 (2.0)**
	Fox	62.2 (1.8)	**65.7 (1.7)**	51.4 (1.9)	54.3 (1.8)
	Tiger	83.0 (1.6)	**85.8 (1.3)**	76.2 (2.0)	80.7 (1.9)
	Elephant	**90.1 (1.0)**	**89.2 (1.0)**	80.3 (1.5)	84.3 (1.4)
	alt.atheism	**70.0 (2.7)**	54.0 (2.7)	57.3 (2.8)	55.0 (2.8)
	rec.motorcycles	**73.7 (2.8)**	54.2 (2.6)	54.8 (2.3)	54.5 (2.8)
	pol.mideast	**70.8 (2.4)**	63.4 (2.3)	61.3 (2.4)	59.7 (2.2)
	Web1	79.9 (2.8)	80.7 (2.8)	75.9 (3.1)	**87.3 (2.5)**
	Web4	**76.4 (2.7)**	62.7 (3.0)	70.6 (3.3)	**70.6 (2.9)**
	Web7	67.8 (2.5)	**74.6 (2.7)**	66.7 (2.5)	**71.7 (2.7)**
	Brown Creeper	67.9 (1.2)	72.6 (1.4)	**85.7 (0.7)**	**85.7 (0.7)**
	Winter Wren	**94.0 (1.0)**	**93.7 (1.1)**	90.7 (1.5)	86.0 (1.8)
	PS Flycatcher	79.5 (1.5)	80.9 (1.1)	**86.1 (0.9)**	**84.5 (1.1)**
Liknon	Musk1	90.9 (1.4)	**93.8 (1.0)**	76.0 (2.4)	79.8 (2.3)
	Musk2	**91.1 (1.8)**	**90.4 (1.8)**	84.6 (2.1)	**90.3 (1.6)**
	Fox	**67.8 (1.7)**	**66.4 (1.6)**	56.4 (2.0)	61.8 (2.1)
	Tiger	**86.7 (1.5)**	**87.9 (1.3)**	83.4 (1.5)	84.9 (1.5)
	Elephant	**90.9 (1.0)**	**90.3 (0.9)**	85.2 (1.5)	88.1 (1.3)
	alt.atheism	**65.5 (2.5)**	56.1 (2.8)	59.0 (2.5)	59.2 (2.6)
	rec.motorcycles	**72.6 (2.5)**	55.0 (2.7)	51.3 (2.4)	50.0 (2.5)
	pol.mideast	**69.1 (2.6)**	**63.8 (2.5)**	50.4 (2.4)	50.2 (2.3)
	Web1	82.1 (2.6)	77.7 (2.6)	81.9 (2.7)	**88.7 (2.0)**
	Web4	**76.7 (3.1)**	55.0 (3.3)	69.8 (2.9)	**72.5 (2.8)**
	Web7	**69.5 (2.9)**	59.8 (3.2)	62.5 (2.6)	**70.4 (2.9)**
	Brown Creeper	**87.1 (0.6)**	**87.6 (0.6)**	85.9 (0.6)	86.0 (0.6)
	Winter Wren	**96.8 (0.4)**	**96.8 (0.5)**	**97.2 (0.3)**	97.0 (0.3)
	PS Flycatcher	**89.5 (0.6)**	**89.2 (0.6)**	86.4 (0.7)	84.6 (0.7)

Table 2. AUC and standard error ($\times 100$) of Winnow and Liknon classifiers, 5×10 cross-validation. Bold indicates results not significantly worse than best per dataset.

The results are shown in Tables 2 and 3. Overall, on these datasets D_{bag} performs the best, followed by D^{inst} and D^{RS}, and D^{BS} in the last place. To ease the comparison, we performed a Friedman rank test[**?**] on each classifier. The ranks are shown in Table 4. Although there are not always significant differences, the ordering of the ranks follows the same pattern in each case.

The fact that D^{bag} and D^{inst} perform comparably (except Newsgroups, as will be explained later) suggests that D^{bag} is able to capture the important information that D^{inst} contains. One conclusion is that real life datasets are less like the Concept, and more like the Distribution dataset from Fig.2. In other words, even the non-concept instances in positive bags may be very informative. Consider images of foxes and tigers. Because foxes and tigers live in a different habitats, the parts of the images containing trees, sand and so forth can tell us something about which animal is probably in the image. Or, as in the bird songs datasets, some birds species can be heard together often.

Although D^{inst} potentially contains more information than D^{bag}, there are few cases where it is a clear winner in terms of performance. It is possible that the low sample size of these datasets limits the full potential of D^{inst}: there are just too many features to deal with. It must be noted that D^{inst} is not completely the

	Dataset	D^{bag}	D^{inst}	D^{BS}	D^{RS}
LIBSVM	Musk1	**93.3 (1.2)**	**93.7 (1.4)**	76.5 (2.4)	80.0 (2.3)
	Musk2	**93.4 (1.4)**	**93.7 (1.3)**	86.5 (2.0)	90.6 (1.7)
	Fox	**67.9 (1.5)**	**67.9 (1.5)**	52.9 (2.0)	60.6 (2.0)
	Tiger	**88.1 (1.5)**	**86.5 (1.4)**	83.2 (1.5)	84.8 (1.5)
	Elephant	**91.3 (0.8)**	89.3 (1.0)	85.6 (1.6)	87.4 (1.3)
	alt.atheism	**70.7 (2.5)**	48.8 (2.7)	49.2 (2.4)	51.1 (2.8)
	rec.motorcycles	**71.4 (2.4)**	42.9 (2.7)	39.0 (2.2)	40.4 (2.5)
	pol.mideast	**69.4 (2.3)**	50.6 (3.0)	52.4 (2.3)	51.8 (2.5)
	Web1	**86.1 (2.4)**	80.2 (2.7)	83.4 (2.1)	**89.6 (1.8)**
	Web4	**75.1 (2.8)**	60.4 (3.2)	**70.7 (3.2)**	**73.7 (2.7)**
	Web7	**71.0 (2.8)**	**68.7 (3.0)**	58.6 (2.8)	**71.6 (2.8)**
	Brown Creeper	**88.0 (0.6)**	**87.6 (0.6)**	84.6 (0.7)	84.2 (0.7)
	Winter Wren	**97.2 (0.4)**	95.7 (0.4)	96.5 (0.4)	96.2 (0.4)
	PS Flycatcher	**89.3 (0.6)**	**89.0 (0.7)**	86.6 (0.7)	84.9 (0.7)
Logistic	Musk1	**92.3 (1.4)**	**91.2 (1.6)**	80.6 (2.4)	83.8 (2.1)
	Musk2	91.4 (1.3)	86.2 (1.8)	91.5 (1.4)	**93.7 (1.1)**
	Fox	**67.1 (1.4)**	**66.4 (1.3)**	60.4 (1.9)	63.5 (1.8)
	Tiger	84.4 (1.5)	**86.3 (1.3)**	**85.0 (1.4)**	**85.4 (1.5)**
	Elephant	**89.2 (0.9)**	**88.8 (1.0)**	86.6 (1.4)	**88.8 (1.2)**
	alt.atheism	**71.3 (2.6)**	55.7 (2.6)	54.2 (2.7)	54.3 (2.7)
	rec.motorcycles	**75.3 (2.5)**	56.2 (2.6)	52.4 (2.6)	50.7 (2.7)
	pol.mideast	**69.9 (2.3)**	61.1 (2.3)	54.1 (2.4)	55.0 (2.2)
	Web1	**87.1 (2.2)**	79.0 (2.9)	78.0 (2.7)	80.4 (2.6)
	Web4	**79.4 (2.9)**	64.3 (3.1)	73.0 (3.3)	73.2 (3.6)
	Web7	69.6 (2.9)	**72.7 (2.8)**	69.3 (3.1)	**75.8 (2.9)**
	Brown Creeper	75.5 (1.0)	80.2 (0.8)	**86.9 (0.6)**	86.6 (0.6)
	Winter Wren	91.5 (0.7)	**94.8 (0.4)**	93.3 (0.4)	91.9 (0.5)
	PS Flycatcher	75.5 (0.9)	80.4 (0.8)	**84.8 (0.7)**	84.4 (0.7)

Table 3. AUC and standard error ($\times 100$) of support vector and logistic classifiers, 5×10 cross-validation. Bold indicates not significantly worse than best per dataset.

same as MILES[5] because there, an exponential similarity function is used which gives more importance to low distances. The effects of such a transformation (on all studied representations) are left for further investigation.

One surprising result is that for Newsgroups, only D^{bag} is able to produce some reasonable performances. One of the reasons is that positive and negative bags contain many instances that are very close together: similar to the background instances in the Concept dataset in Fig.2. However, a few true positive instances are very far away from all other instances, and from each other. There are so few of them, that even the sparse classifiers using D^{inst} are not able to select only the correct ones. However, they are so far away from everything, that they can still sufficiently influence the dissimilarities in D^{bag}. The asymmetry of D^{bag} also plays a role here; by transposing D^{bag}, much better results can be achieved [6]. However, this is not as straightforward for D^{inst}, so we did not pursue this possibility here.

It is interesting that D^{RS} often performs better than D^{BS}. To find out why, we examined the performances of the individual classifiers of both ensembles. Some typical results are shown in Fig.3. It is, indeed, often the case that D^{RS} produces more accurate classifiers. One reason for this could be the dimensionalities per classifier: D^{BS} can often have classifiers built on just 2 or 3 dimensional spaces (especially for Musk, Fox, Tiger and Elephant). However, sometimes D^{BS} and D^{RS} have classifiers with similar performances, but the ensemble using D^{RS}

Classifier	D^{bag}	D^{inst}	D^{BS}	D^{RS}	F	CV	Reject H_0?	CD
Winnow	2.14	2.36	2.85	2.64	0.82	2.85	No	-
Liknon	1.50	2.50	3.29	2.71	6.48	2.85	Yes	1.25
LIBSVM	1.29	2.64	3.36	2.71	10.86	2.85	Yes	1.25
Logistic	2.14	2.36	3.07	2.43	1.38	2.85	No	-

Table 4. Ranks of Friedman test (best possible is 1, worst is 4). The null hypothesis H_0 is that there are no significant differences between ranks. H_0 is rejected (significance of 5%) when the F-value of the ranks is larger than the critical value (CV). Significant differences are those larger than the critical difference (CD).

is still much better. This suggests that classifiers built on each bag separately provide more correlated information, than classifiers built on random selections of instances.

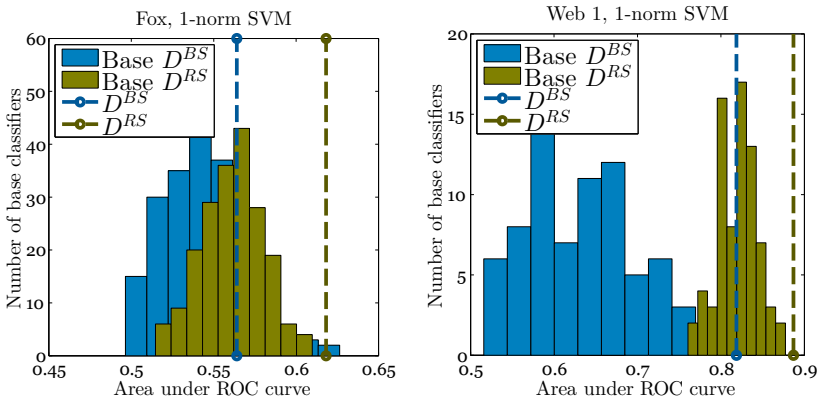

Fig. 3. Histograms of performances of individual base classifiers and the final ensembles of D^{BS} and D^{RS} for the Fox and Web Recommendation 1 datasets.

A way to improve the performance of an ensemble is to use a trained combiner which would learn which individual classifier outputs most often correspond with the true labels of the training set. This could filter out the less accurate classifiers from the ensemble, increasing the overall performance. Looking at Fig.3, we would expect such performance improvements to be possible, especially for D^{BS}. Following [8], we have performed a few experiments with the nearest mean combiner, both on D^{BS} and D^{RS}. The results, however, were quite disappointing: for both ensembles, only minor improvements, if any, could be achieved. A possible cause for this is that nearest mean combiner was trained on normalized posterior probabilities, while the original classifier outputs might have been more informative.

We have found that for D^{BS}, there is little relation between the label of the prototype and the performance of the classifier. This is in line with the idea that positive and negative bags may not have the same type of background instances. On the other hand, we have found medium to strong correlations between dimensionality and the AUC of the individual classifiers. It might be worth investigating whether this can help us to select more informative prototypes a priori, before creating the dissimilarity matrix. Furthermore, there might be room for improvement for D^{RS}. The subspaces are allowed to be larger than the average number of instances per bag, because the dissimilarities are sampled with replacement.

6 Conclusion

We examined several dissimilarity representations for Multiple Instance Learning. These representations are based on distances of bags to prototype bags or instances. We investigated how such distances can be combined in order to create informative dissimilarities, and how this affects the dimensionality of the final representation. We considered combining such distances by averaging, by concatenating or by ensembling subspace classifiers, where each classifier is trained on a selection of the instance distances.

Averaging instance distances into a bag-based representation reduces the dimensionality and performs very well. Although the concatenated, instance-based representation is potentially the most informative, its rather high dimensionality might be harmful for performance. Lower dimensionality can also be achieved by combining subspace classifiers. However, in this case it is more difficult to achieve good performances because more variables, such as subspace size and the combining rule, are involved. It remains a question how to create and select such informative subspaces.

The bag representation produces good results, which means that averaging does not dilute the information of the individual instances. This suggests that in practice, most instances in a bag can be informative. In other words, the distributions of instances from positive and negative bags may be very different in general, and not only in terms of the presence or absence of a concept. A reasonable conclusion is that in such cases, it is better to use the bag representation, which requires less resources but still provides good performances.

Acknowledgements We would like to acknowledge the financial support of the ACM-W/Microsoft scholarship.

References

1. Andrews, S., Hofmann, T., Tsochantaridis, I.: Multiple instance learning with generalized support vector machines. In: National Conference on Artificial Intelligence. pp. 943–944 (2002)

2. Bhattacharyya, C., Grate, L., Rizki, A., Radisky, D., Molina, F., Jordan, M., Bissell, M., Mian, I.: Simultaneous classification and relevant feature identification in high-dimensional spaces: application to molecular profiling data. Signal Processing 83(4), 729–743 (2003)
3. Briggs, F., Lakshminarayanan, B., Neal, L., Fern, X., Raich, R., Hadley, S., Hadley, A., Betts, M.: Acoustic classification of multiple simultaneous bird species: A multi-instance multi-label approach. J. Acoust. Soc. of America 131, 4640 (2012)
4. Chang, C., Lin, C.: Libsvm: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2(3), 27 (2011)
5. Chen, Y., Bi, J., Wang, J.: Miles: Multiple-instance learning via embedded instance selection. Pattern Analysis and Machine Intelligence 28(12), 1931–1947 (2006)
6. Cheplygina, V., Tax, D., Loog, M.: Class-dependent dissimilarity measures for multiple instance learning. Structural, Syntactic, and Statistical Pattern Recognition pp. 602–610 (2012)
7. Dietterich, T., Lathrop, R., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artificial Intelligence 89(1-2), 31–71 (1997)
8. Duin, R., Tax, D.: Experiments with classifier combining rules. Multiple Classifier Systems pp. 16–29 (2000)
9. Foulds, J., Frank, E.: A review of multi-instance learning assumptions. Knowledge Engineering Review 25(1), 1 (2010)
10. Foulds, J., Frank, E.: Revisiting multiple-instance learning via embedded instance selection. AI 2008: Advances in Artificial Intelligence pp. 300–310 (2008)
11. Ho, T.: The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(8), 832–844 (1998)
12. Huang, J., Ling, C.: Using auc and accuracy in evaluating learning algorithms. Knowledge and Data Engineering, IEEE Transactions on 17(3), 299–310 (2005)
13. Kittler, J.: Combining classifiers: A theoretical framework. Pattern Analysis & Applications 1(1), 18–27 (1998)
14. Littlestone, N.: Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. Machine learning 2(4), 285–318 (1988)
15. Maron, O., Lozano-Pérez, T.: A framework for multiple-instance learning. In: Advances in neural information processing systems. pp. 570–576. Morgan Kaufmann Publishers (1998)
16. Pękalska, E., Duin, R.: On combining dissimilarity representations. Multiple Classifier Systems pp. 359–368 (2001)
17. Pękalska, E., Duin, R.: The dissimilarity representation for pattern recognition: foundations and applications, vol. 64. World Scientific Pub Co Inc (2005)
18. Pękalska, E., Duin, R., Paclík, P.: Prototype selection for dissimilarity-based classifiers. Pattern Recognition 39(2), 189–208 (2006)
19. Tax, D., Duin, R.: Learning curves for the analysis of multiple instance classifiers. Structural, Syntactic, and Statistical Pattern Recognition pp. 724–733 (2008)
20. Tax, D., Loog, M., Duin, R., Cheplygina, V., Lee, W.: Bag dissimilarities for multiple instance learning. Similarity-Based Pattern Recognition pp. 222–234 (2011)
21. Weidmann, N., Frank, E., Pfahringer, B.: A two-level learning method for generalized multi-instance problems. Machine Learning: ECML 2003 pp. 468–479 (2003)
22. Zhou, Z., Jiang, K., Li, M.: Multi-instance learning based web mining. Applied Intelligence 22(2), 135–147 (2005)
23. Zhou, Z., Sun, Y., Li, Y.: Multi-instance learning by treating instances as non-iid samples. In: Int. Conf. on Machine Learning. pp. 1249–1256. ACM (2009)