Handling Label Noise in Video Classification via Multiple Instance Learning

Thomas Leung[1], Yang Song[1], and John Zhang[2*]
[1] Google Inc., Mountain View, CA 94043
[2] Columbia University, New York, NY 10027
{leungt,yangsong}@google.com, jrzhang@cs.columbia.edu

Abstract

In many classification tasks, the use of expert-labeled data for training is often prohibitively expensive. The use of weakly-labeled data is an attractive solution but raises the problem of label noise. Multiple instance learning, whereby training samples are "bagged" instead of treated as singletons, offers a possible approach to mitigating the effects of label noise. In this paper, we propose the use of MILBoost [28] in a large-scale video taxonomic classification system comprised of hundreds of binary classifiers to handle noisy training data. We test on data with both artificial and real-world noise and compare against the state-of-the-art classifiers based on AdaBoost. We also explore the effects of different bag sizes on different levels of noise on the final classifier performance. Experiments show that when training classifiers with noisy data, MILBoost provides an improvement in performance.

1. Introduction

The growth of multimedia data on the Internet provides great challenges and opportunities for computer vision applications such as image and video classification. One challenge is the lack of expert labeled training data, especially for videos [12, 30]. Obtaining high-quality training data has always been a central issue for machine learning tasks. With the growing number of categories (up to hundreds of thousands) in the spectrum, the problem is more severe than ever. However opportunities exist: there are weakly labeled images and videos available, together with a large amount of unlabeled data. In this paper we focus on the use of weakly labeled data.

Weakly labeled data on the Internet can come from different sources, e.g., images and videos associated with some search queries [4, 11, 15], or videos labeled by amateur raters without full knowledge of the categories in consideration. In weakly labeled data, some instances may be labeled with the wrong classes. While it is tempting to employ these data, handling class label noise becomes a critical issue.

One intuitive way to handle label noise is to pre-process the data, meaning that data are filtered before feeding to classifier training. Different strategies have been tried along this line. For example, for videos and images associated with a search query, we only keep the top-ranked one. Another strategy is to pick data with high confidence scores if there are pre-trained weak classifiers available. Incremental learning is used in [15, 19]. Clustering techniques can also be used to select data consistent with their peers. These different methods essentially try to pick data with high confidence. However, a good confidence measure is not always available, and the selected high-confidence data can lead to distribution bias in the training data. Another way to handle label noise is post-processing, applied after a round of classifier training. Training data can be partitioned into smaller sets, and learned models are combined to alleviate label noise, as done in [29]. The post-processing strategy may impose additional computational cost if multiple classes are needed in classification.

This paper proposes a novel scheme to handle label noise. It tackles the problem from the center—handling noise within classifier training. We use the idea of Multiple Instance Learning (MIL) to tackle label noise. It is naturally embedded in the classifier training process. It does not suffer from the aforementioned drawbacks of pre-processing or post-processing approaches. Furthermore, if desired, it can be easily combined with other strategies since it is inherent in classifier training.

In our approach, Multiple Instance Learning principle is applied to learning a boosting-based classifier [28]. The idea of using MILBoost to alleviate label noise is novel. While learning is based on MIL principle, the resulting classifiers preserve all the advantages of AdaBoost. AdaBoost has been very popular and performs well in multiple applications. Our approach can be directly applied to any problems where AdaBoost is used, but with the additional benefit of training label noise being automatically handled.

We choose a large-scale video classification problem as

[*] Work done as an intern at Google.

the application domain to demonstrate the effectiveness of our algorithm. One reason for choosing videos is that expert labels for videos are especially time consuming to obtain due to the nature of online videos[24, 30]. Our work is one of the first efforts (if not the first) in handling labeling noise in such a large scale video classification system.

Worthy of mention is that attribute (or feature) noise and class label noise are two sources of noise for a learning algorithm [31]. There has been much attention on reducing feature noise, such as using principal component analysis (PCA), but less systematic work in handling label noise. Our work sheds new lights in this direction.

This paper is organized as follows. Related works are reviewed in Section 2. Section 3 presents our proposed scheme for reducing label noise using MILBoost. In Section 4, we describe a large scale video taxonomic classification system, where our approach for handling label noise is applied. Section 5 depicts our data collection. In Section 6, experiments are performed using both synthetic noise and a noisy training set to illustrate real-world challenges. Finally, we conclude and discuss possible directions for future work in Section 7.

2. Related Work

2.1. Multiple Instance Learning

In Multiple Instance Learning (MIL), samples are not treated as positive or negative singletons, but instead grouped together into "bags". Bags are labeled positive if they contain at least one positive sample or negative otherwise. MIL has been successfully applied to computer vision tasks including face detection [13, 28], pose detection [2], image categorization [7], segmentation [25], and tracking [3].

Andrews and Hofmann [1] propose the use of linear programming in AdaBoost in an MIL approach to reduce the effects of label noise. However, as the authors remark, "Our current implementation could not handle large data sets". That is because it uses a linear programming step in the inner loop of the boosting algorithm, thus making it prohibitively slow with large data sets.

2.2. Removing Label Noise

The increasing popularity of using weakly-labeled data has inspired works in handling label noise in training sets. One way to handle label noise is to apply Latent Dirichlet Allocation (LDA) [5], probabilistic Latent Semantic Analysis (pLSA) [14, 23] or its extensions, and remove unrelated latent topics [4, 11]. However, it faces the issue of how to select good topics and how many topics to keep. In [11], an extra validation set is used to select one best topic for classification. The choice of keeping only one topic is heuristic/empirical, and it may limit the benefits of latent topic models. In [4], the relevant topics are selected manually.

Incremental learning is used in [15, 19] to select relevant training samples out of noisy candidates. As noted in [19] and also observed in our own experience, a brute-force application of incremental learning is likely to select samples similar to previous rounds, and therefore resulting in biased models. This issue is addressed in [19] by selecting image samples with high entropy with respect to the latent topics, in addition to the high-likelihood criterion.

We don't intend to claim our proposed method is superior to the above work. Rather, it is a new approach to tackle the label noise problem from a different perspective, at different stages of classifier training. Our method has similar spirit to [26] in applying the idea of Multiple Instance Learning to classifier training. [26] can be looked at as an augmentation of SVM, but ours is a boosting based approach. We also study the problem of video classification, as opposed to image classification.

Another attempt to remove noise in boosting is proposed in [16], where samples with very high weights are regarded as noise and removed from the training set. While intuitive in concept, as admitted by the authors, the effectiveness of the method depends on the selection of threshold for removing samples. In practice, the method is very brittle and impractical with the large variability of video data. We show our experiments using [16] in Section 6.1.

3. MILBoost For Label Noise Removal

We use Multiple Instance Learning in a boosting framework to handle label noise. In most MIL applications to computer vision, the bags are formed very naturally. For example, in object recognition, labels are associated to an image. The presence of the object is known while its exact location within the image is not. Each bag then contains all the segments within the image. MIL is used to figure out which segment corresponds to the object during training. In a object tracking problem, each bag will contain the windows around all possible locations to where the object has moved. Again, MIL is used to obtain the best location.

In the problem of label noise removal, training examples are given as independent singletons. Some of the training examples are correctly labeled, while others are not. Each positive MIL bag is formed by multiple positive training examples, randomly and uniformly selected from the positive set. The negative examples are singletons, or in other words, the negative MIL bags have a size of 1. The reason for this asymmetry is that in most practical applications, the noise level in the negative set is negligible. Imagine the problem of learning a classification model for the animal "lamb". The first two pages of results from Google Image Search using the query "lamb" contain only 36 real pictures of lamb, out of a total of 60 images. The noise level in the positive set is thus 40%. On the other hand, the chance of any random image from the Web containing a picture of a "lamb" is minimal. A high level overview of our approach is depicted in Figure 1.

We follow the MILBoost formulation of Viola et al [28]. Each example resides in a bag. We use the index i to represent the bags and j for samples within the bag. The probability of an example being positive is the logistic function:

$$p_{ij} = \frac{1}{1 + exp(-y_{ij})} \tag{1}$$

where $y_{ij} = H(x_{ij})$. $H(x_{ij})$ is the strong classifier and is a weighted sum of weak classifiers $H(x_{ij}) = \sum_t \lambda_t h_t(x_{ij})$.

We adopt the Noisy OR model for each bag. The probability that a bag is positive is given by

$$p_i = 1 - \prod_{x_{ij}} (1 - p_{ij}) \tag{2}$$

This Noisy OR model essentially means that as long as one sample in each bag is a true positive, the bag will be positive. In other words, even if all other samples in the bag are mislabels, the effect of noise is minimized in classifier training. For example, if we *know* that there is a 50% chance a positive training sample is mislabeled, using a bag size of 2 reduces noise level to 25%[1]. Using bag size of 4, 6, and 8 would reduce noise levels to 6.2%, 1.6%, and 0.4% respectively.

We minimize the negative log likelihood:

$$L(H) = -\sum_{i \in +} log(p_i) - \sum_{i \in -} log(1 - p_i) \tag{3}$$

of all the samples.

In many applications, there could be an imbalance in the number of positive and negative samples. It is usually much easier to gather negative training samples than positive ones. For example, there are fewer videos about "Comics & Animation" than otherwise. To tackle this imbalance, we weigh the two terms in Eq. 3 differently and redefine the loss to be

$$L(H) = -\alpha \sum_{i \in +} log(p_i) - \sum_{i \in -} log(1 - p_i) \tag{4}$$

Following standard derivation of boosting weights, the weight of sample j in positive bag i is

$$w_{ij}^{+} = \alpha \frac{1 - p_i}{p_i} p_{ij} \tag{5}$$

The weight for samples in the negative bags is

$$w_{ij}^{-} = p_{ij} \tag{6}$$

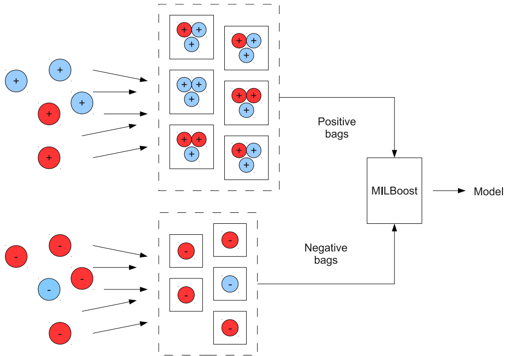

Figure 1. Overview of our approach. True positive (blue +) and mislabeled positive (red +) sample videos are grouped as positive bags, while negative samples (red −, including mislabeled negative samples, blue −) are kept as negative singletons. Together, they are used to train a MILBoost model.

Each boosting round is to find the weak classifier $h(x)$ that minimizes

$$\sum_{i \in +} \sum_{j} h(x_{ij}) w_{ij}^{+} - \sum_{i \in -} \sum_{j} h(x_{ij}) w_{ij}^{-} \tag{7}$$

The resulting strong classifier is $H(x) = \sum \lambda_t h_t(x)$. We find λ_t using line search to minimize $L(H + \lambda_t h_t)$.

In our implementation, α is computed so that the total initial weights of all positive samples is equivalent to the total initial weights of the negative ones. Our line search algorithm is n-ary search followed by parabolic fitting.

From a practical point of view, the input samples and model output format are exactly the same as standard AdaBoost. Moreover, the training and testing speed of our system is essentially the same as standard AdaBoost. Training each category requires on the order of minutes after features are precomputed. The time to classify a video is the same as Adaboost classifiers. In our implementation, with features pre-computed, classifying one video takes less than 1 millisecond. The speed for video feature extraction (Section 4.2) is near-real-time on a modern PC.

4. Video Classification

We demonstrate the effectiveness of our approach on a large-scale video taxonomic classification system. The video classification system assigns categories to videos from a pre-defined taxonomy [24]. The taxonomy is a tree of depth-5, with 1037 nodes. Each node corresponds to a category. A binary classifier is trained for each category in the taxonomy. We adopt the hierarchical one-against-all approach. For a given category, the positive samples are those belonging to the category itself and its descendants; the negative samples are the rest excluding those belonging to the ancestor categories. Classification decisions are based upon results from individual classifiers. Multiple labels (categories) can be assigned to one video. Text-based

[1] This assumes the samples in each bag are uncorrelated and the mislabeling probability is independent.

features and video-content-based features are used for classification. A boosting-based classifier with decision stumps as weak classifiers is used for each binary classification.

4.1. Text-Based Feature Extraction

There are two steps in text-feature extraction. In the first step, weighted text clusters are obtained from video meta-data (title, description and keywords) through Noisy-Or Bayesian Networks [21]. In the second step, pre-trained (using labeled web documents) text-based linear SVM ([10]) classifiers are applied to weighted text clusters and a classification score is obtained for each category. These scores are concatenated into a vector and treated as text features [24]. The second step exploits the knowledge embedded in the text-based classifiers learned from labeled web documents, therefore the features obtained are more effective, especially when the number of training videos is small [24].

4.2. Video-Content-Based Feature Extraction

Video content-based visual and audio features are extracted from each video. Visual features include histogram of local features, color features, texton histograms, edge features, face features, and motion features.

Histograms of oriented gradients (HOG) feature: At each pixel location, we extract a 1800-dimensional feature descriptor, which is the concatenation of 18-dimentional-HOG [8] in a 10 by 10 surrounding window. The raw descriptors are then collected into a bag-of-words representation by quantizing them using a randomized decision tree similar to [22]. This tree is binary with 10 levels, but not full, and has 647 leaves. The bags-of-words accumulate across all frames, at a certain temporal down-sampling rate.

The following features are also computed for each frame (with down-sampling): *Hue-Saturation color histogram*, *texton histogram* [18] with vocabulary size 1000, *edge features* including fraction of edge pixels and edge direction histogram, and *face features* [27] with the number of faces and the ratio of largest face area to the image area. Vector quantization is performed on each type of feature descriptors, and histograms are accumulated throughout the video.

Cuboid interest point detector [9] is used to extract *motion features*. Spatio-temporal volumes are extracted around the detected interest points. Two types of descriptors are used to represent each cuboid. First, normalized pixel values are concatenated and principal component analysis (PCA) is applied to reduce the dimensionality to 256. Second, each slice of the cuboid is split into 2 by 2 cells, and all HOG descriptors of these cells in the cuboid are concatenated into a vector. Similar to the first type, PCA is applied. Both descriptors are further quantized using their corresponding codebooks.

In addition to visual features, the following two audio features are extracted: mel-frequency cepstral coefficients (MFCC) [6] and stabilized auditory images (SAI) [20].

5. Data

We have obtained training data by asking human raters to assign categories to videos. The raters are given a video and asked to provide labels after watching (part of) it. There are two types of raters: *experts* and *amateurs*.

5.1. Expert Raters

Expert raters have extensive training and familiarity with the taxonomy. Their task is to label videos through direct text input, *i.e.*, to pick one or more labels out of the whole taxonomy of 1037 categories. They have demonstrated high inter-rater agreement and are thus regarded as *experts*. These data are referred to as ground-truth videos since they are of the highest quality, with little or no label noise. All the testing data in our experiments come from this ground truth set. The disadvantages of collecting data using experts are obvious: it is extremely time-consuming and expensive.

5.2. Amateur Raters

Due to the difficulty in obtaining ground-truth videos, we explore using amateur raters to provide video labels through a Mechanical Turk-like environment. These raters have no prior training and are not familiar with the taxonomy. The labeling task is reduced to ten binary questions whereby each rater is asked to indicate whether or not a given category applies to the video. The raters have the option of choosing none. These candidate categories are selected using text-based classifiers discussed in Section 4. The ten most confident categories are taken, randomly shuffled, and presented to the raters.

We ask five raters to independently label each video, and we require at least 3 raters to agree on a label before retaining it. In order to examine the labeling accuracy, a validation experiment has been performed where 1,000 ground-truth videos are presented to amateur raters to label. The results of the validation experiment are summarized in Table 1. From Table 1, (1 - precision) is roughly the percentage of noisy labels. Thumbnails of sample videos from this validation experiment are shown in Figure 2.

Amateur raters have advantages in time, speed and cost. The amateur raters are able to label 25922 videos within one month, whereas the experts require approximately one year to label 10528 ground-truth videos. Amateur raters require little or no supervision, and no training beyond the brief instructions presented to them as part of the labeling user interface. The experts, on the other hand, receive considerable training and practice in order to become familiar with the taxonomy. The expert raters usually have to be paid higher than the amateur raters as well.

Figure 2. Sample videos from amateur raters. Those with blue frames are mislabeled compared to ground-truth labels. The text string below each video thumbnail is its VIDEO_ID. The video can be watched via link http://www.youtube.com/watch?v=VIDEO_ID. Videos in the first row are labeled as */Arts & Entertainment/Music & Audio/Pop Music* by amateur raters, and videos in the second row are labeled as */Autos & Vehicles/Custom & Performance Vehicles*. Ground truth labels for the mislabeled videos are as follows, EdbGy-fkk4E: */Arts & Entertainment/Music & Audio/Urban & Hip-Hop/Rap & Hip-Hop*, */Arts & Entertainment/Music & Audio/Dance & Electronic Music*; 16FeMFD4uxA: */Arts & Entertainment/Music & Audio/Urban & Hip-Hop/Soul & R&B*, */Hobbies & Leisure/Special Occasions/Holidays & Seasonal Events/Christmas*; lDTVLY6oi0w: */Arts & Entertainment/Movies/Musical Films*, */People & Society/Kids & Teens*; PBIjn8NHV-I: */Arts & Entertainment/Music & Audio/Rock Music*, */Hobbies & Leisure/Special Occasions/Holidays & Seasonal Events/Christmas*; Up13DWbLARs: */Autos & Vehicles/Vehicle Brands/Buick*, */Autos & Vehicles/Vehicle Shows*, */Shopping/Luxury Goods*; XNxxnj50h0Q: */Autos & Vehicles/Vehicle Brands/Honda*.

Depth	# Labels	Precision	Recall	F-score
1	1762	74	55	63
2	1871	66	42	51
3	1214	61	36	45
4	177	59	39	46
5	11	83	45	58

Table 1. Performance of amateur raters when asked to label 1000 ground-truth videos. Each video is labeled by 5 raters independently, and only labels with at least 3 raters agreed are retained. We believe the high precision at depth 5 is mainly due to the small sample size.

6. Experiments

Section 5 describes how our data is obtained. There are 10528 videos labeled by expert raters and 25922 videos labeled by amateur raters. The candidate videos are randomly selected from YouTube videos. Experiments in Section 6.1 uses 80% of the expert labeled videos as training data, the other 20% as testing data. Experiments in 6.2 use data from amateur raters as training data, and data from expert raters for testing.

6.1. Effect of Noise on Video Classification

We use the expert labeled data for this experiment. To study the effect of noise on classification performance, we switch some of the negative training samples to be positive training samples. We create data sets with 0%, 10%, 20%, 50%, and 80% noise in the positive set. In most practical applications, the noise level in the negative set is usually negligible. Therefore, we do not add noise to the negative set. We also investigate the effect of different bag sizes, namely, 2, 4, and 8.

Figure 3 shows the results for the categories "Arts & Entertainment", "Pop Music", and "Pets & Animals". These categories belong to different depth in the taxonomy and have large differences in the number of ground truth videos. We compare MILBoost with different bag sizes to standard AdaBoost. In low noise levels (0% and 10%), MILBoost performs comparably to AdaBoost, particular for small bag sizes (2 or 4). The advantage of MILBoost is significant when noise level is high. Similar improvements are observed for categories with different number of ground truth labels.

We also compare the performance using the standard equal error rate, *i.e.*, the error rate at which false positive rate equals to false negative rate. The equal error rates averaged over 77 categories trained are shown in Table 2. (Each of these 77 categories has at least 100 ground truth videos.) In essence, there is no significant performance difference at low noise levels (0% and 10%). At 20% noise or above, MILBoost shows performance gain. At 80% noise, the gain is more than 5% with bags of 8.

Table 3 shows the equal error rate of a handful of randomly selected categories at different depth levels of the taxonomy. As expected, there is a lot of variations in performance gain/loss across categories. Overall, at low noise levels (below 10%), there is either no difference or a slight degradation with MILBoost. At higher noise levels, MILBoost clearly wins.

The amount of noise we experiment with might seem excessive (50% or 80%). However, if we want to take advantage of the large amount of image or video data available on the Internet, this amount of noise level is to be ex-

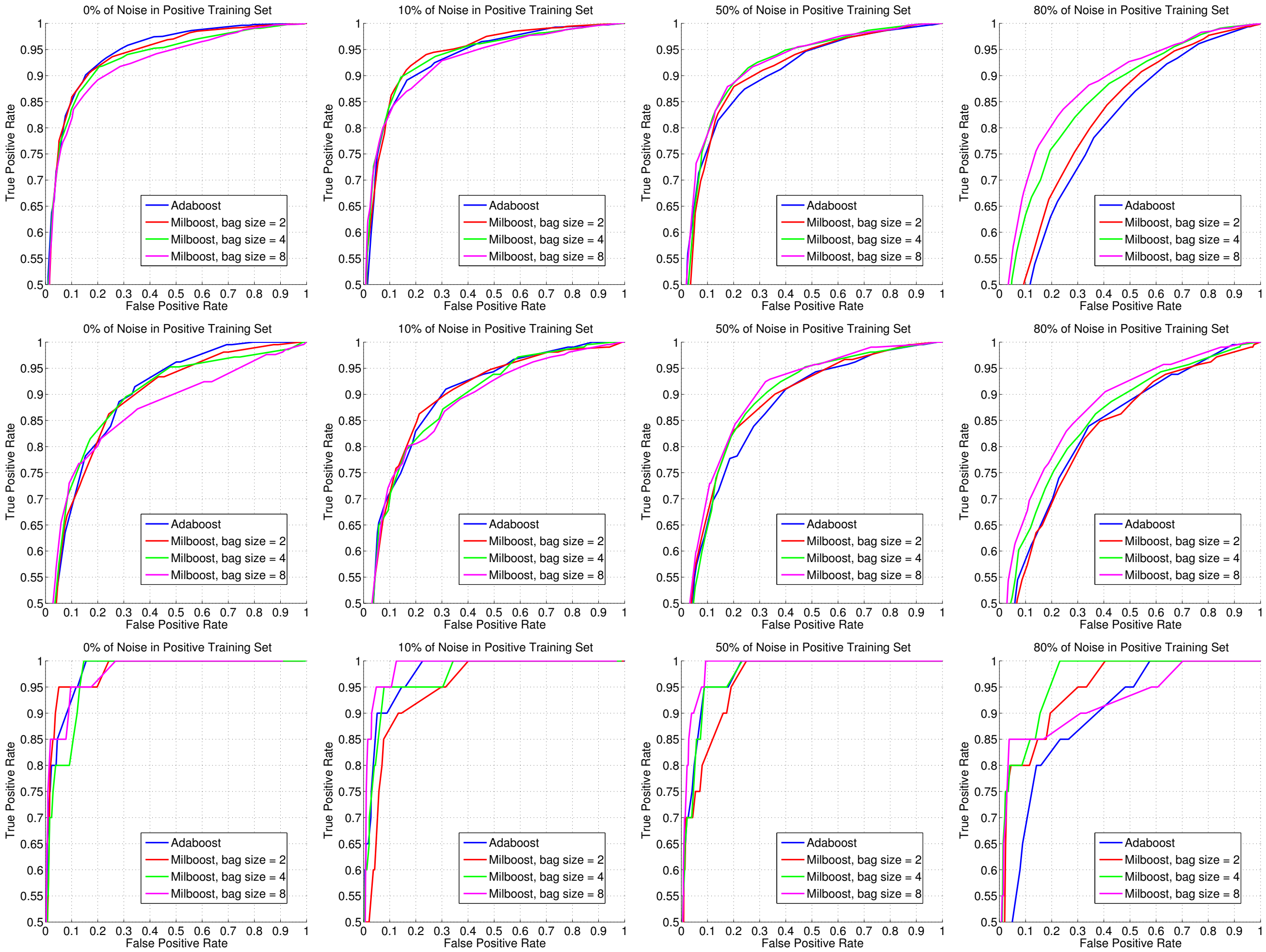

Figure 3. ROC curves on the video categorization problem. The three rows are results for categories "Arts & Entertainment" (5235 ground truth videos), "Pop Music" (685 videos), and "Pets & Animals" (90 videos) respectively. We compare the performance of MILBoost with different bag sizes to AdaBoost. Each column show the performance at different noise levels: 0%, 10%, 50%, and 80%.

Noise	AB	MB (2)	MB (4)	MB (8)
0%	**13.5%**	14.0%	14.6%	14.1%
10%	14.5%	15.3%	14.8%	**14.0%**
20%	15.4%	15.2%	15.1%	**13.9%**
50%	17.5%	17.6%	16.3%	**15.5%**
80%	21.9%	21.0%	18.9%	**16.7%**

Table 2. Performance comparison of AdaBoost (AB) and MILBoost (MB) averaged over all 77 trained categories. We use the standard equal error rate for comparison (the lower the better). As can be seen, MILBoost compares favorably with AdaBoost. The performance gain is significant when noise level is high, achieving over 5% gain at 80% noise with bag size of 8.

pected. It is observed by Fergus et al [11] that images obtained from Google Image Search typically contain upward of 85% label noise. In general, for any large scale classification problem with thousands or millions of categories, collecting clean, noise-free data is impractical.

The effect of noise on the video taxonomy classification task is also evaluated using the method proposed in [16]. [16] treats samples with weights larger than a threshold as noise and discards them. We follow the authors' suggestion of exhaustively trying thresholds between 3 and 20 and use cross validation to select the appropriate value. However, the method is very brittle. Even at 0% noise, in all but 2 categories, a majority of the positive samples are eventually treated as noise and discarded, resulting in very poor classification performance. We believe this is due to the large variability of the video data.

6.2. Experiments on Real-world Noisy Data

In this section, we demonstrate the effects of MILBoost on a real-world application with a naturally noisy training set. Video data from amateur raters (Section 5.2) are used as training set. As shown in Table 1, the data are quite

Noise	AB	MB (2)	MB (4)	MB (8)
Pets & Animals (1)				
0%	8.9%	**5.1%**	11.6%	9.0%
10%	9.5%	11.6%	7.3%	**5.0%**
50%	8.0%	13.3%	8.3%	**6.7%**
80%	18.3%	15.0%	**14.1%**	15.0%
Games (1)				
0%	11.7%	13.4%	15.3%	**11.2%**
10%	14.0%	12.7%	13.1%	**12.3%**
50%	19.9%	17.0%	13.8%	**12.6%**
80%	18.4%	18.9%	18.6%	**16.7%**
Custom & Performance Vehicles (2)				
0%	4.3%	7.1%	4.8%	**3.7%**
10%	7.2%	8.4%	6.0%	**4.3%**
50%	8.8%	8.0%	**6.5%**	7.2%
80%	13.0%	15.8%	10.6 %	**6.5%**
Pop Music (3)				
0%	19.4%	19.5%	**18.0%**	19.9%
10%	18.9%	**18.0%**	19.1%	19.5%
50%	21.6%	18.5%	18.6%	**18.4%**
80%	24.5%	25.4%	22.9%	**21.1%**

Table 3. Performance Comparison of AdaBoost (AB) and MILBoost (MB) on 4 randomly selected categories. We use the equal error rate to compare the performance. At 0 to low noise levels, MILBoost does not improve the performance by much, if at all. At higher noise levels, MILBoost has a clear advantage. These categories are chosen so that they span different depth levels in the taxonomy and also different number of ground truth videos. For the 4 categories, the number of positive ground truth videos are 90, 390, 250, 685 respectively.

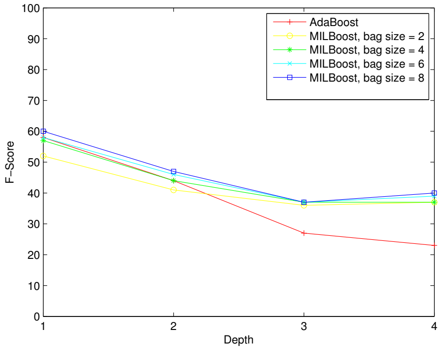

Figure 4. Comparison of classification performance between MILBoost with bag sizes 2, 4, 6, 8, as well as AdaBoost.

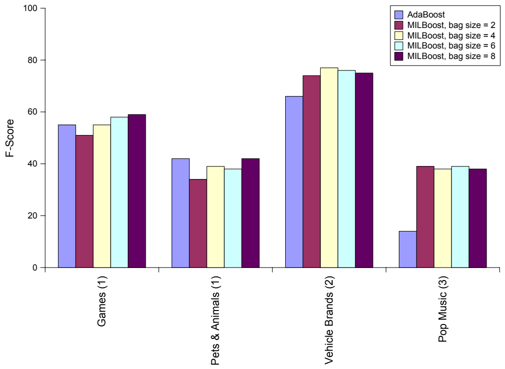

Figure 5. Comparison of classification F-score between MILBoost with bag sizes 2, 4, 6, 8, as well as AdaBoost for four categories. Depths of categories are shown in parentheses.

noisy, up to 40% error. From these data, there are 224 categories with at least 40 videos. MILBoost and AdaBoost based classifiers are trained for those categories. MILBoost classifiers are trained using 2, 4, 6 and 8 samples per bag. Evaluation is performed on ground-truth video data from Section 5.1. Performance is quantified by the standard definitions of precision, recall and F-score.

Figure 4 gives an overview of classifier performance. Each data point in Figure 4 is the F-score averaged over categories in the corresponding taxonomy depth level. The average performance of the classifiers trained using MILBoost with 8 samples per bag exceeds that of all others trained using noisy data, including AdaBoost.

To get an in-depth look, we examine the classification performance for four categories: “Games”, “Pets & Animals” (both depth-1), “Vehicle Brands” (depth-2), and “Pop Music” (depth-3). The changes in F-score are shown in Figure 5. From our validation experiments evaluating rater accuracy (Section 5.2), we can get a sense of the amount of noise present in these categories. Three of the categories examined (“Games”, “Pets & Animals”, “Vehicle Brands”) are relatively clean with high precision in the validation experiments (80%, 77%, 75% respectively), but “Pop Music” had a very high amount of label noise with a precision of 23%. Sample video thumbnails from this category are shown in the first row of Figure 2. As described in the caption of Figure 2, videos from categories like */Arts & Entertainment/Music & Audio/Urban & Hip-Hop/Rap & Hip-Hop* (the top left video, with VIDEO_ID being EdbGyfkk4E) can be mislabeled as category */Arts & Entertainment/Music & Audio/Pop Music*. The significant increase in performance through the use of MILBoost over AdaBoost for this category reinforces our observation that MILBoost offers significant performance gains when applied to sets containing high amounts of noise.

7. Conclusion

Label noise is a part of life in classification problems. It is amplified when data from the Internet are used. In this paper, we propose a method to reduce the effect of label noise within classifier training using principles of Multiple Instance Learning. In particular, we use the formulation of

MILBoost and put multiple positive samples in a bag. We systematically analyze the effect of bag size on classification performance through artificial noise generation. Our algorithm works exceptionally well in the presence of large amount of noise. We also demonstrate a practical application where we use amateur raters to collect lots of noisy data. Our proposed algorithm shows improved performance over traditional learning by AdaBoost.

There are several directions that we will continue to explore. First, we will study how to automatically determine bag sizes. As our experiments have shown, a large bag size (8) improves the classification performance at high noise levels, while it does not have a significantly adverse effect on low noise levels. However, on some problems where noise is not sigificant, it is possible that bigger bags will descrease classification performance. Cross-validation is a possible solution for bag size selection. Second, there are more sources of training videos. Videos from text classifiers (Section 4.1) can be used directly as noisy training examples. Third, we will investigate the use of multiple bag sizes in the same classifier learning system. Different data sources, or videos with different levels of rater agreement, have different noise levels. Different bag sizes can take advantage of the varying noise levels. Fourth, we will explore using other MIL loss functions in the boosting framework. In this paper, the Noisy OR model is used. Other models, such as average probability, or the ISR criterion in [17], might yield better performance.

References

[1] S. Andrews and T. Hofmann. Multiple-instance learning via disjunctive programming boosting. In *In NIPS*, 2004. 2

[2] B. Babenko, P. Dollár, Z. Tu, and S. Belongie. Simultaneous learning and alignment: Multi-instance and multi-pose learning. In *Faces in Real-Life Images*, October 2008. 2

[3] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In *CVPR*, 2009. 2

[4] T. L. Berg and D. A. Forsyth. Animals on the web. In *Proc. CVPR*, 2006. 1, 2

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. of Machine Learning Research*, 2003. 2

[6] G. Chechik, E. Ie, M. Rehn, S. Bengio, and D. Lyon. Large-scale content-based audio retrieval from text queries. In *Proceeding of the 1st ACM international conference on Multimedia information retrieval*, pages 105–112. ACM, 2008. 4

[7] Y. Chen, J. Bi, and J. Z. Wang. Miles: Multiple-instance learning via embedded instance selection. *PAMI*, 2006. 2

[8] N. Dalal and B. Triggs. Histogram of oriented gradients for human detection. In *Proc. CVPR*, 2005. 4

[9] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *ICCV'05 Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2005. 4

[10] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *J. of Machine Learning Research*, 2008. 4

[11] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning object categories from google's image search. In *Proc. ICCV*, 2005. 1, 2, 6

[12] R. Fergus, Y. Weiss, and A. Torralba. Semi-supervised learning in gigantic image collections. In *NIPS*, 2009. 1

[13] M. Guillaumin, J. Verbeek, and C. Schmid. Multiple instance metric learning from automatically labeled bags of faces. In *Proc. ECCV*, 2010. 2

[14] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, 1999. 2

[15] N. Ikizler-Cinbis, R. G. Cinbis, and S. Sclaroff. Learning actions from the web. In *Proc. ICCV*, 2009. 1, 2

[16] A. Karmaker and S. Kwek. A boosting approach to remove class label noise. In *Proc. International Conference on Hybrid Intelligent Systems*, pages 206–211, 2005. 2, 6

[17] J. D. Keeler, D. E. Rumelhart, and W.-K. Leow. Integrated segmentation and recognition of hand-printed numerals. In *NIPS*, 1990. 8

[18] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *IJCV*, 43(1), 2001. 4

[19] L.-J. Li, G. Wang, and L. Fei-Fei. Optimol: automatic object picture collection via incremental model learning. In *Proc. CVPR*, 2007. 1, 2

[20] R. Lyon, M. Rehn, S. Bengio, T. Walters, and G. Chechik. Sound retrieval and ranking using sparse auditory representations. *Neural computation*, 22(9):2390–2416, 2010. 4

[21] R. E. Neapolitan. *Learning Bayesian Networks*. Prentice-Hall, Inc., 2003. 4

[22] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *Proc. CVPR*, 2008. 4

[23] J. Sivic, B. C. Russell, A. Efros, A. Zisserman, and W. T. Freeman. Discovering object categories in image collections. In *Proc. ICCV*, 2005. 2

[24] Y. Song, M. Zhao, J. Yagnik, and X. Wu. Taxonomic classification for web-based videos. In *CVPR*, 2010. 2, 3, 4

[25] A. Vezhnevets and J. Buhmann. Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning. In *Proc. CVPR*, 2010. 2

[26] S. Vijayanarasimhan and K. Grauman. Keywords to visual categories: Multiple-instance learning for weakly supervised object categorization. In *Proc. CVPR*, 2008. 2

[27] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *Proc. CVPR*, 2001. 4

[28] P. Viola, J. C. Platt, and C. Zhang. Multiple instance boosting for object detection. In *NIPS*, 2006. 1, 2, 3

[29] Z. Wang, et. al. Youtubecat: Learning to categorize wild web videos. *CVPR*, 2010. 1

[30] S. Zanetti, L. Zelnik-Manor, and P. Perona. A walk through the web's video clips. In *Proc. 1st IEEE Workshop on Internet Vision*, 2008. 1, 2

[31] X. Zhu, X. Wu, and Q. Chen. Eliminating class noise in large datasets. In *Proc. ICML*, 2003. 2