YouTubeCat: Learning to Categorize Wild Web Videos

Zheshen Wang[1], Ming Zhao[2], Yang Song[2], Sanjiv Kumar[3], and Baoxin Li[1]

[1]Computer Science and Engineering, Arizona State University, Tempe, AZ 85281, USA
[2]Google Research, Mountain View, CA 94043, USA
[3]Google Research, New York, NY 10011, USA

[1]{zheshen.wang, baoxin.li}@asu.edu
[2,3]{mingzhao, yangsong, sanjivk}@google.com

Abstract

Automatic categorization of videos in a Web-scale unconstrained collection such as YouTube is a challenging task. A key issue is how to build an effective training set in the presence of missing, sparse or noisy labels. We propose to achieve this by first manually creating a small labeled set and then extending it using additional sources such as related videos, searched videos, and text-based webpages. The data from such disparate sources has different properties and labeling quality, and thus fusing them in a coherent fashion is another practical challenge. We propose a fusion framework in which each data source is first combined with the manually-labeled set independently. Then, using the hierarchical taxonomy of the categories, a Conditional Random Field (CRF) based fusion strategy is designed. Based on the final fused classifier, category labels are predicted for the new videos. Extensive experiments on about 80K videos from 29 most frequent categories in YouTube show the effectiveness of the proposed method for categorizing large-scale wild Web videos[1].

1. Introduction

On-line services for archiving and sharing personal videos such as YouTube have become quite popular in recent years. Automatic categorization of videos is important for indexing and search purposes. However, it is a very challenging task for such a large corpus of practically unconstrained (wild Web) videos. A lot of efforts have been devoted to video analysis in the past, but most existing works use very limited number of videos or focus on specific domains such as news, sports etc. Due to practically unbounded diversity of Web videos in both content and quality (as illustrated in Figure 1), analysis of such data is much more challenging than relatively clean videos expected by most existing techniques. A recent study by Zanetti et al. showed that most existing algorithms did not perform well on general Web videos [25]. It also pointed out that one of the major challenges in Web video categorization is the lack of sufficient training data. Manually labeling videos is both time-consuming and labor intensive – on one hand one has to watch part of a video before (s)he can suggest labels; on the other, web videos are extremely diverse in nature, thus even for human experts, summarizing the video content by using a few keywords is not an easy task.

Figure 1. Examples of wild YouTube videos showing extremely diverse visual content.

In this paper, we propose a novel approach that combines multiple data sources for wild YouTube video categorization. Starting from a small number of manually labeled samples (as few as 50 per category), we expand the training set by propagating labels to their co-watched videos, collecting data by using internet video search engines (such as Google video search), and even incorporating data from other domains (e.g., text-based webpages). These additional data sources are first pairwise combined with manually-labeled data and a classification model is trained for each combination. For fusing these trained mod-

[1]This work was performed when the first author interned at Google.

els, we propose a CRF-based tree-DRF fusion approach, which views the taxonomy tree as a random field. Each node (i.e. a category) is associated with a binary label and the output likelihoods of the trained models (applied on the training data) are used as local observations for the nodes. Unlike a traditional fusion strategy that treats each category independently, tree-DRF makes the final labeling decision as a whole by explicitly taking the hierarchical relationships among the categories into consideration. This is crucial to achieve good performance since the data from additional sources is usually quite noisy. The hierarchical relationships among categories provides powerful context for alleviating the noise. Results from extensive experiments on 80K YouTube videos demonstrate that the proposed solution outperforms existing methods that either use just a single data source or traditional data fusion strategy.

The main contributions of this work can be summarized as follows: First, to the best of our knowledge, this is the first work that deals with categorization of unconstrained Web videos at such a large scale. Second, we propose a novel approach for integrating data from multiple disparate sources for classification given insufficient training data. Finally, we introduce a tree-DRF based fusion strategy that exploits the hierarchical taxonomy over categories and effectively deals with noise in multiple data sources. It significantly outperforms other commonly used fusion strategies based on SVM and iterative co-training [2, 3, 8].

The rest of the paper is organized as follows. We first review the related literature in Section 2 followed by the description of multiple data sources we use in Section 3. The proposed solution with pairwise data combination and tree-DRF based fusion strategy is presented in Section 4. Extensive experimental results, comparisons and analysis are reported in Section 5. We conclude in Section 6 with a brief discussion on future work.

2. Related Work

Compared to image analysis, research on video analysis has been relatively recent. Most existing approaches are either limited to some specific domains (e.g. movies [4, 12], TV videos [5, 21, 24] etc.) or focus on certain predefined content such as human face [5, 19] and human activities [14]. However, large scale categorization of wild Web videos still remains an unsolved problem. The works of Schindler et al. [20], VideoMule [17] and Zanetti et al. [25] are among the initial efforts in this direction. Schindler et al. tried video categorization on 1500 user uploaded videos from 15 categories using bag-of-words representation. However, the classification performance is very poor on this general video set (best classification accuracy is 26.9%). Ramachandran et al. proposed VideoMule, a consensus learning approach to multi-label YouTube videos classification using YouTube categories. Specific amount of data and categories were not reported in their work.

Zanetti et al. explored existing video classification methods on about 3000 YouTube videos in their recent work [25]. They pointed out that a major difficulty in Web video analysis is the lack of enough labeled training data. Semi-supervised machine learning approaches [27] are useful for expanding training data in general. However, graph-based methods used commonly for semi-supervised learning e.g., [28] and semi-supervised SVM [1] are inefficient for large amounts of data with high-dimensional features. Popular co-training/self-training approaches [2, 3, 8] are also typically expensive and their performance is quite sensitive to the amount and quality of the initial training set.

Another possible way of collecting more training data is to make use of data from other sources including different domains. It is worth noting that combining multiple data sources is more challenging than combining multiple views of the same data [2, 3, 8], since properties of different data sources are typically more diverse. Multiple data sources can be combined with either early fusion or late fusion strategies [22]. Typically, early fusion assumes that all the features are available for each video, which is not valid in our case (e.g. webpage data has only text features). In late fusion, classifier models are first trained separately; then the trained models are applied to the training set. At the fusion stage, obtained likelihoods from different models are concatenated for each sample and used as a feature vector. Another round of training is then carried out on the new 'features'. Traditional fusion methods are based on regular learning algorithms (such as SVM, AdaBoost), which treat each category independently. On the contrary, given a hierarchical taxonomy over categories, it is desirable to exploit such relationships to achieve robust classification. In this paper, we propose tree-DRF to handle the category structure while doing late fusion and empirically show the benefits of such approach.

3. Multiple data sources

As mentioned earlier, lack of labeled training data is a main bottleneck for general Web video categorization. To alleviate this problem, we first manually labeled 4345 videos from all the 29 categories as initial seeds. This set is further expanded by including samples from related videos, searched videos and cross-domain labeled data (i.e. text webpages), as illustrated in Figure 2. Details of each data source are given below.

3.1. Manually-labeled data

To collect the initial seeds for training, we first build a category taxonomy with the help of professional linguists. About 1000 categories are defined using a hierarchical tree of 5 vertical levels (Depth-0 to Depth-4 from top to bottom,

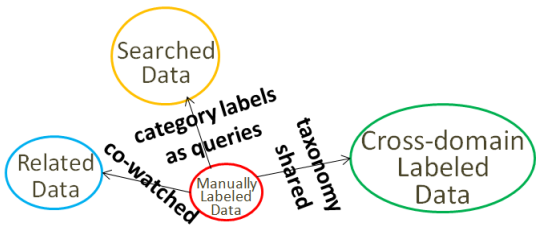

Figure 2. Multiple data sources for YouTube videos including a small set of manually labeled data, related (e.g. co-watched video data), searched data collected by using a video search engine with categories as queries, and cross-domain data (e.g. webpages) which are labeled with the same taxonomy structure.

Depth-0 is the root). Randomly selected YouTube videos that have been viewed more than a certain number of times are labeled by professionally-trained human experts based on the established taxonomy. Each video is labeled from Depth-0 to the deepest depth it can go. For example, if a video is labeled as *Pop Music*, it must be associated with label *Music & Audio* and *Art & Entertainment* as well. Note that this is a general taxonomy instead of being designed for YouTube videos specifically. Thus, it is not surprising that the distribution of manually-labeled videos over all categories is extremely unbalanced. For example, the *Art & Entertainment* category contains close to 90% of all the labeled videos, and categories such as *Agriculture & Forestry* have only a few videos. In fact, such imbalance reflects the real distribution of videos in the entire YouTube corpus. In this paper, we work on 29 categories that had a reasonable amount of manually-labeled samples, i.e., more than 200 for Depth-1 categories and more than 100 for Depth-2 to 4 categories. Manually-labeled samples from these 29 categories (4345 samples in total) cover close to 80% of all the data we labeled, roughly implying that the categories we are working with cover $\sim$80% of all possible videos on YouTube. To the best of our knowledge, this is the first paper which deals with general Web video classification on such diverse categories. In our experiments, 50% randomly selected samples are used as initial seeds for training (denoted as "M") and the remaining 50% are used for testing.

3.2. Related (Co-watched) data

To increase the training samples for each category, we considered co-watched videos, i.e., the next videos that users watched after watching the current video. We empirically noticed if a video is co-watched more than 100 times with a certain video, they tend to have the same category. Of course, such labels can be noisy but our tree-DRF based late fusion method is able to handle such noise robustly. So, in our experiments, co-watched videos (denoted as "R") of all the initial seed videos with co-watch counts larger than 100 (3277 video in total) are collected to assist training.

3.3. Searched data

Another possibility for expanding the training set is by searching for videos using online video search engines using a category label as a text query. For example, returned videos by submitting a query "soccer" may be used as training samples for the "soccer" category. Constrained by the quality of existing search engines, searched videos may be noisy. In our work, we keep about top 1000 videos returned for each category. Since the categories form a hierarchical structure, the videos returned for categories at lower levels are included for their ancestors as well. Querying Google video search gave us a set of about 71,029 videos (denoted as "S").

3.4. Cross-domain labeled data

Compared to video labeling, assigning labels to other types of data (e.g. text-based webpages) is usually easier. Although such data comes from a completely different domain, it can be helpful for video classification as long as the samples are labeled using the same taxonomy. This is because we also use text-based features to describe each video as explained in Section 4.1. We collected 73,375 manually-labeled webpages (denoted as "W") as one of the additional data sources in our experiments.

4. Learning from multiple data sources

In Section 3, in addition to the manually-labeled data, we introduced several auxiliary sources which may be useful for boosting the video classification accuracy. The main challenge is how to make use of such diverse set of data with different properties (e.g., video content features are not available for webpages) and labeling quality (e.g., labels of searched and co-watched data are fairly noisy).

In this paper, we propose a general framework to integrating data from mixed sources. As illustrated in Figure 3, each auxiliary data source is first pairwise combined with the manually-labeled training set. Initial classifiers are trained on each such pair. For each pair, two separate classifiers are learned, one with text-based and another with content-based features. For example, in Figure 3, M_{Sc} is a content-based and M_{St} is a text-based model for the combination of manually-labeled data and searched data. Trained models are then fused using a tree-DRF fusion strategy. Different from traditional methods that fuse models for each category independently, the proposed tree-DRF incorporates the hierarchical taxonomy structure exploring the category relationships effectively.

Next we introduce the features used for training individual classifiers followed by the description of our tree-DRF fusion method.

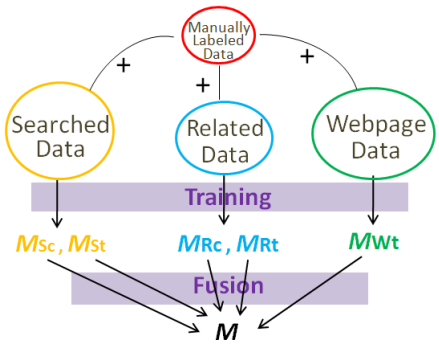

Figure 3. General framework of the proposed solution: Additional data sources are first combined with manually-labeled data independently and classifier models are trained based on either text or content features for each combination. Individual classifier are further fused to form the final classifier M.

4.1. Features

It is well known that designing good features is perhaps the most critical part of any successful classification approach. To capture the attributes of wild Web videos as completely as possible, state-of-the-art text and video content features are utilized in our experiments as briefly summarized below.

Text features: For each video, the text words from title, description and keywords are extracted. Then, all these words are weighted to generate text clusters. The text clusters are obtained from Noisy-Or Bayesian Networks [16], where all the words are leaf nodes in the network and all the clusters are internal nodes. An edge from an internal node to a leaf node means the word in the leaf node belongs to that cluster. The weight of the edge means how strongly the word belongs to that cluster.

Video content features: *color histogram* computed using hue and saturation in HSV color space, *color motion* defined as cosine distance of color histograms between two consecutive frames, *skin color* features as defined in [9], *edge features* using edges detected by Canny edge detector in regions of interest, *line features* using lines detected by probabilistic Hough Transform, *histogram of local features* using Laplacian-of-Gaussian (LoG) and SIFT [15], *histogram of textons* [13], *entropy features* for each frame using normalized intensity histogram and entropy differences for multiple frames, *face features* such as number of faces, size and aspect ratio of largest face region (faces are detected by an extension of AdaBoost classifier [23]), *shot boundary* detection based features using difference of color histograms from consecutive frames [26], *audio features* such as audio volume and 32-bin spectrogram in a fixed time frame centered at the corresponding video frame, *adult content features* based on a boosting-based classifier in addition to frame-based adult-content features [18]. We extract the audio and visual features in the same time interval. Then, a 1D Haar wavelet decomposition is applied to them at 8 scales. Instead of using the wavelet coefficients directly, we take the maximum, minimum, mean and variance of them as the features in each scale. This multi-scale feature extraction is applied to all our audio and video content features except the histogram of local features [7].

Note that features are not the main contribution of this work. Due to space limitation, we skip the details of the features and refer the reader to the respective references. For fair comparisons, all the experimental results reported in this work are obtained based on the same set of features.

4.2. CRF-based fusion strategy

Conditional Random Fields (CRFs) are graph-based models that are popularly used for labeling structured data such as text [11] and were introduced in computer vision by [10]. In this work, we use outputs of discriminative classifiers to model the potentials in CRFs as suggested in Discriminative Random Field (DRF) formulation in [10]. Following the notation in [10], we denote the observations as $\boldsymbol{y}$ and the corresponding labels as $\boldsymbol{x}$. According to CRFs, the conditional distribution over labels given the observations is defined as a Gibbs field:

$$p(\boldsymbol{x}|\boldsymbol{y}) = \frac{1}{Z}(\sum_{i \in S} A_i(x_i, \boldsymbol{y}) + \sum_{i \in S} \sum_{j \in N_i} I_{ij}(x_i, x_j, \boldsymbol{y})), \quad (1)$$

where S is the set of all the graph nodes, N_i is the set of neighbors of node i, and Z is a normalizing constant called partition function. Terms A_i and I_{ij} are the unary and pairwise potentials sometimes referred to as *association potential* and *interaction potential* respectively [10].

4.3. Tree-DRF

As discussed earlier, in this work we use multiple data sources that are combined by a late fusion step. We want a fusion strategy that can combine the classifier outputs from different sources while respecting the taxonomy over categories. The DRF framework described above gives a natural way of achieving that. Formally, A_i learns to fuse the outputs of independent classifiers while I_{ij} enforces the category relationships defined by the hierarchical taxonomy.

In [10], DRF is used for image classification, in which a graph is built on image entities, i.e., pixels or blocks. On the contrary, in our case, the graph is defined over the hierarchical taxonomy (i.e., a tree over categories) and a node represents a category. Each node i is associated with a binary label variable x_i, i.e., $x_i \in \{-1, 1\}$ implying whether i^{th} category label should be assigned to the input video or not. The scores from different classifiers for the i^{th} category on a given video are concatenated in a feature vector, which serve as the observation y_i. Figure 4 illustrates the proposed tree-DRF.

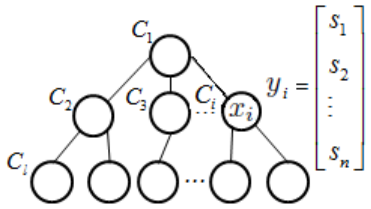

Figure 4. Late fusion strategy based on tree-DRF. For each input video, a tree-structure over categories is defined. The binary label at the i^{th} node (x_i) represents whether that video should be assigned the category label C_i. The observation vector (y_i) is simply the concatenation of classifier scores on the video for that category.

Following [10], *association potential* is defined as,

$$A_i(x_i, \boldsymbol{y}) = \log \frac{1}{1 + exp(-x_i \boldsymbol{w}_i^T h_i(\boldsymbol{y}))}, \qquad (2)$$

where $\boldsymbol{w}_i$ is a parameter vector and $h_i(\boldsymbol{y})$ is a feature vector at site i. Following [10], we define $h_i(\boldsymbol{y})$ to include the classifier scores and their quadratic combinations.

Note that unlike the homogeneous form used in [10], the association potential in our tree-DRF model is inhomogeneous. There is a separate association parameter $\boldsymbol{w}$ for each node. The reason is that since a different set of classifiers is learned for each category (i.e, a node), forcing the weight vectors defining combinations of such disparate sets of classifiers to be the same for all the nodes is too harsh. Thus, we allow the model to chose a different weight vector for each category. Of course, it leads to more parameters in the model but since our graph is fairly small (just 29 nodes), and the size of observation vector, i.e., the number of classifiers, is also small, the computational overhead was negligible. Moreover, overfitting is also not a concern since we have enough training data for such small number of parameters.

The *interaction potential* in tree-DRF is defined as,

$$I_{ij}(x_i, x_j, \boldsymbol{y}) = x_i x_j \boldsymbol{v}^T \mu_{ij}(\boldsymbol{y}), j \in N_i, \qquad (3)$$

where $\boldsymbol{v}$ are the model parameters and $\mu_{ij}(\boldsymbol{y})$ is a pairwise feature vector for nodes i and j. In this work, we only explored data-independent smoothing by forcing $\mu_{ij}(\boldsymbol{y})$ to be a constant. Similarly, the parameter $\boldsymbol{v}$ was kept to be the same for all the node pairs. One can easily relax this to allow directional (anisotropic) interactions between parents and children which can provide more powerful directional smoothing. We plan to explore this in the future.

We used the standard maximum likelihood method for parameter learning in tree-DRF. Since the graph structure is a tree, exact unary and pairwise marginals were computed using Belief Propagation (BP). For inference, we used site-wise Maximum Posterior Marginal (MPM), again using BP. Results of tree-DRF fusion and comparisons to regular fusion strategy based on SVM and Co-training are presented in Section 5.

5. Experiments and results

In order to verify the effectiveness of the proposed solution, we performed extensive experiments with about 80K YouTube videos and about 70K webpages. We first introduce the experimental data and settings in the next section followed by a brief description of the evaluation metric.

5.1. Experimental data and setting

As described in Section 3, four different data sources and 29 major categories are used in our experiments. The categories followed by their path in the taxonomy tree are: "Arts & Entertainment" (1), "News" (2), "People & Society" (3), "Sports" (4), "Celebrities & Entertainment News" (1, 5), "Comics & Animation" (1, 6), "Events and Listings" (1, 7), "Humor" (1, 8), "Movies" (1, 9), "Music & Audio" (1, 10), "Offbeat" (1, 11), "Performing Arts" (1, 12), "TV & Video" (1, 13), "Team Sports" (4, 14), "Anime & Manga" (1, 6, 15), "Cartoons" (1, 6, 16), "Concerts & Music Festivals" (1, 7, 17), "Dance & Electronic Music" (1, 10, 18), "Music Reference" (1, 10, 19), "Pop Music" (1, 10, 20), "Rock Music" (1, 10, 21), "Urban & Hip-Hop" (1, 10, 22), "World Music" (1, 20, 23), "TV Programs" (1, 13, 24), "Soccer" (4, 14, 25), "Song Lyrics & Tabs" (1, 10, 19, 26), "Rap & Hip-Hop" (1, 10, 22, 27), "Soul & R&B" (1, 10, 22, 28), and "TV Reality Shows" (1, 13, 24, 29).

In our experiments, binary classifiers are trained for each category respectively. Content features and text features are trained separately by using AdaBoost and SVM, respectively. LibLinear [6] is used to train SVMs when training samples exceed 10K. Trained models are then integrated using regular SVM based late fusion strategy [22]. Since webpage data has only text features (no content features), only a single model is learned for this set. The training data from two sources (i.e., manually-labeled data plus one additional data source) is combined before training the classifiers. After all the data sources are leveraged, fusion is performed for content and text features for three pairwise combinations, represented by five individual classifiers. In the training process, negative training samples for each category are randomly selected from other categories with a negative-positive ratio of 3:1.

5.2. Evaluation metrics

While testing, since binary classifiers are trained for each category, each test sample receives 29 classification decisions (either "yes" or "no"). Multiple labels for a single sample are allowed. As the category labels form a taxonomy structure, predicted categories/labels are also propagated to their ancestors as done while generating ground-truth labels for the training data. For example, if a test sample has a ground-truth label "Art & Entertainment" / "TV & Video" / "TV Programs", it is treated as a true positive

Table 1. Classification accuracy of each data source, including manually labeled data (M), related data (R), searched data (S) and webpage data (W). Webpage data achieved the best performance except for Depth-2.

F-score	Depth-1	Depth-2	Depth-3	Depth-4
M	0.80	**0.60**	0.45	0.41
R	0.74	0.53	0.37	0.34
S	0.73	0.51	0.37	0.31
W	**0.84**	0.54	**0.48**	**0.45**

sample for "Art & Entertainment" category if it is classified by any of these three classifiers. For the quantitative evaluation, we compute Precision, Recall and F-score. To perform aggregate assessment of the classification performance, we also compute F-scores for each depth level of the taxonomy.

5.3. Results and analysis

The objective of the proposed approach is to improve video classification performance by making use of data from multiple sources of varied quality. Table 1 lists classification accuracy of each data source (due to space limitation, we only show F-score in all tables and figures). Performance with just the related videos (R) or the searched videos (S) is much worse than that from manually-labeled data (M). It shows that neither related videos or searched videos are sufficient for training a reliable classifier. Webpage data (W) obtained from a completely different domain, which does not even contain video content, works better than manually-labeled data for most taxonomy depths. This is possible since even noisy text based features for videos are usually more reliable than video content features.

In order to achieve better results, we combine each of the additional data sources pairwise with manually-labeled training data. As shown in Table 2, for related video source, pairwise combination achieves significant improvements over just using related videos and even better than training on manually-labeled data. For the searched videos, performance of pairwise combination is also better than that for just the searched data, but worse than that of the manually-labeled data. In terms of the webpage data, pairwise combination is not always superior to the single sources. Overall, there are two observations: 1) Pairwise combination with manually-labeled data can improve classification accuracy of any single additional source in most cases; 2) Introducing additional data sources by simply merging them with the manually-labeled data does not guarantee improvement for all cases over the baseline configuration, i.e., using just the manually-labeled data for training.

Next, we fuse the single classifier models trained from pairwise combinations to further boost the classification performance. First row of Table 3 shows the results of using regular SVM late fusion strategy. Compared to the best

Table 2. Classification accuracy of each combination of manually-labeled data with one additional data source. The combination with related data achieves significant improvements over just using the related data and even outperforms using only manually-labeled data. But the later observation is not true for the other two cases (i.e. combination with searched data or webpage data).

F-score	Depth-1	Depth-2	Depth-3	Depth-4
M + R	**0.86**	**0.63**	**0.47**	**0.49**
M + S	0.78	0.57	0.43	0.37
M + W	0.84	0.55	0.45	0.39

Table 3. Classification accuracy of fusing pairwise combinations of data using different fusion strategies. The proposed tree-DRF approach outperforms any single data source or their pairwise combinations. It is also superior to the traditional SVM fusion strategy with the same features.

F-score	Depth-1	Depth-2	Depth-3	Depth-4
All, SVM	0.84	0.65	0.46	0.49
All, Tree-DRF	**0.87**	**0.72**	**0.57**	**0.52**
M+R, Tree-DRF	0.85	0.66	0.48	0.45

cases in Table 2, fusing all data sources does not achieve any obvious improvement (for Depth-1 and Depth-3, results are even worse). It is because, for SVM, when the feature dimension increases but not the amount of training data, the test performance may degenerate due to overfitting. This observation underscores our previous assertion that an inappropriate fusion strategy for adding unreliable data sources may even harm the classification accuracy.

Results of the proposed tree-DRF fusion strategy are reported in Table 3-second row. For all taxonomy depths, tree-DRF outperforms regular SVM fusion. Especially for Depth-2 and Depth-3, in which the categories can benefit from both parent categories and child categories, it achieves 0.07 (11%) and 0.11 (24%) improvements in F-scores. Compared to the baseline performance (Table 1-first row), it gains 0.07 (9%), 0.12 (20%), 0.12 (27%), 0.11 (27%) F-score improvements for Depth-1 to Depth-4 respectively. Such significant improvements are due to the taxonomy tree based learning of tree-DRF. In other words, since interactions between parent and child nodes are considered, noise in the additional data sources can be largely filtered. This is because useful information is typically consistent for neighboring nodes and thus can be emphasized by the *interaction potential* in tree-DRF.

For analyzing the effectiveness of including additional data sources, we applied tree-DRF on the pair of manually-labeled data and related data (which gave the best results among all pairwise combinations with regular fusion of content models and text models) in the third row of Table 3. Compared to tree-DRF on all data (second row in Table 3), results are worse, which demonstrates the gain from multiple data sources by using tree-DRF. For easy comparison,

accuracies from all experiments are summarized in Figure 5.

To analyze the results for individual categories, we illustrate F-scores for the baseline method (i.e., using only manually-labeled data for training), and SVM and tree-DRF based fusion with all data sources in Figure 6. For most of the categories, tree-DRF outperforms the other two methods, especially for the categories with small amount of training samples but relatively large number of neighbors.

In addition to SVM and tree-DRF based fusion, we also conducted experiments with co-training on different combinations of the four data sources with different settings (e.g. by varying the number and weights of new training samples added in each iteration, and the stopping criteria). In the best case, F-scores for Depth-1 to Depth-4 were 0.82, 0.61, 0.44 and 0.40 respectively, which are much lower than the proposed tree-DRF method and even lower than regular SVM fusion strategy.

Regarding computational complexity of tree-DRF, since the graph is built on the taxonomy, it results in a very small graph having just 29 nodes connected with very sparse edges. Also, since the outputs of individual classifiers are used as features, it leads to very low-dimensional features. Hence, overall the tree-DRF is extremely fast in training as well as testing.

6. Conclusion and future work

In this paper, we proposed a novel solution to wild web video categorization on a large-scale dataset (more than 80 thousand YouTube videos). Our approach provides an effective way of integrating data from diverse sources, which largely alleviates a major problem of lack of labeled training data for general web video classification. Tree-DRF was proposed for fusing models trained from individual data sources when combined with small amount of manually-labeled data in a pairwise fashion. Compared to traditional fusion strategies, the proposed tree-DRF takes the taxonomy tree of category labels into account, resulting in significant improvement in classification performance. Experimental results on a large-scale YouTube dataset show that the proposed approach is effective for categorizing wild videos on the Web.

Currently we only consider undirected relationships between parent and child categories in tree-DRF. More sophisticated anisotropic formulations of *interaction potential* for parent or child neighbors, and siblings may further improve the labeling performance. In addition, it is also possible to make use of unsupervised learning methods (e.g. clustering) for assigning weights to noisy labeled samples and adjusting their contributions accordingly while training classifiers. Integrating an iterative co-training framework of incrementally adding additional unlabeled data is also a possible way of further expanding the training data set and improving the classification performance.

References

[1] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[2] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. of Workshop on Computational Learning Theory*, 1998.

[3] C. M. Christoudias, R. Urtasun, A. Kapoor, and T. Darrell. Co-training with noisy perceptual observations. In *Proc. of CVPR*, 2009.

[4] O. Duchenne, I. Laptev, J. Sivic, F. Bach, and J. Ponce. Automatic annotation of human actions in video. In *Proc. of ICCV*, 2009.

[5] M. Everingham, J. Sivic, and A. Zisserman. Hello! my name is... buffy automatic naming of characters in tv video. In *Proc. of BMVC*, 2006.

[6] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[7] U. Gargi and J. Yagnik. Solving the label–resolution problem in supervised video content classification. In *Proc. of ACM Multimedia Information Retrieval*, 2008.

[8] S. Gupta, J. Kim, K. Grauman, and R. Mooney. Watch, listen & learn: Co-training on captioned images and videos. In *Proc. of ECML*, 2008.

[9] M. J. Jones and J. M. Rehg. Statistical color models with application to skin detection. *IJCV*, 46(1):81–96, 2002.

[10] S. Kumar and M. Hebert. Discriminative fields for modeling spatial dependencies in natural images. In *Proc. of NIPS*, 2003.

[11] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of ICML*, 2001.

[12] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Proc. of CVPR*, 2008.

[13] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *IJCV*, 43(1):29–44, 2001.

[14] J. Liu, J. Luo, and M. Shah. Recognizing realistic actions from videos. In *Proc. of CVPR*, 2009.

[15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[16] R. E. Neapolitan. *Learning Bayesian Networks*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2003.

[17] C. Ramachandran, R. Malik, X. Jin, J. Gao, K. Nahrstedt, and J. Han. Videomule: a consensus learning approach to multi-label classification from noisy user-generated videos. In *Proc. of ACM MM*, 2009.

[18] H. A. Rowley, Y. Jing, and S. Baluja. Large scale image-based adult-content filtering. In *Proc. of VISAPP*, 2006.

[19] M. E. Sargin, H. Aradhye, P. J. Moreno, and M. Zhao. Audiovisual celebrity recognition in unconstrained web videos. In *Proc. of ICASSP*, 2009.

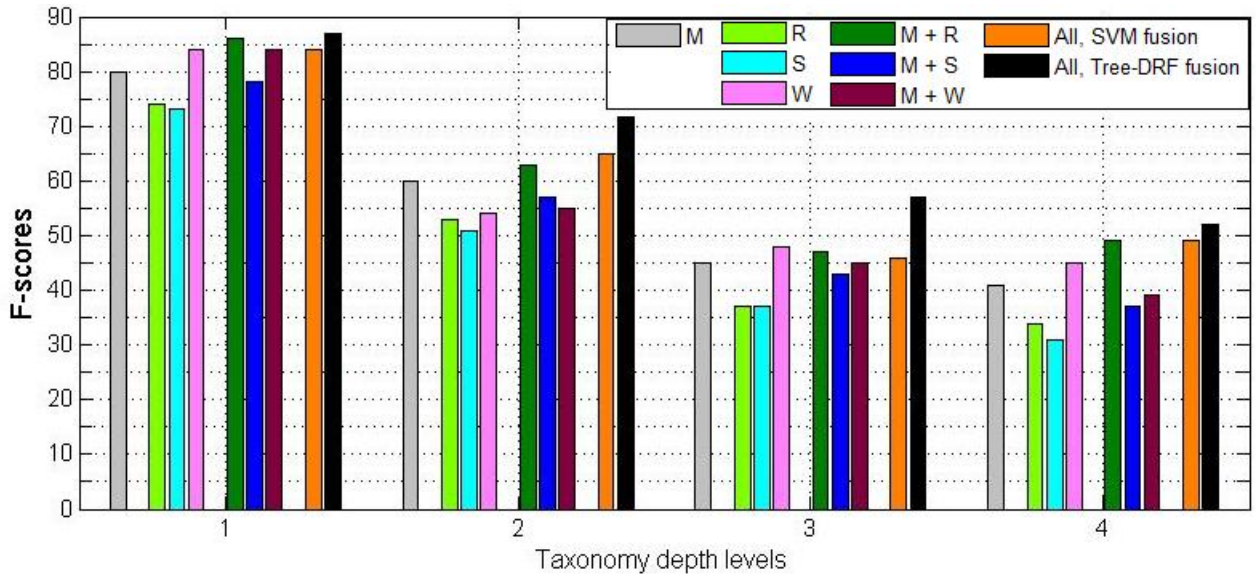

Figure 5. Comparison of classification accuracies from different data sources and combinations. Tree-DRF with all pairwise data combinations achieved the best performance. M: Manually-labeled data, R: Related Videos, S: Searched Videos, W: Webpage data.

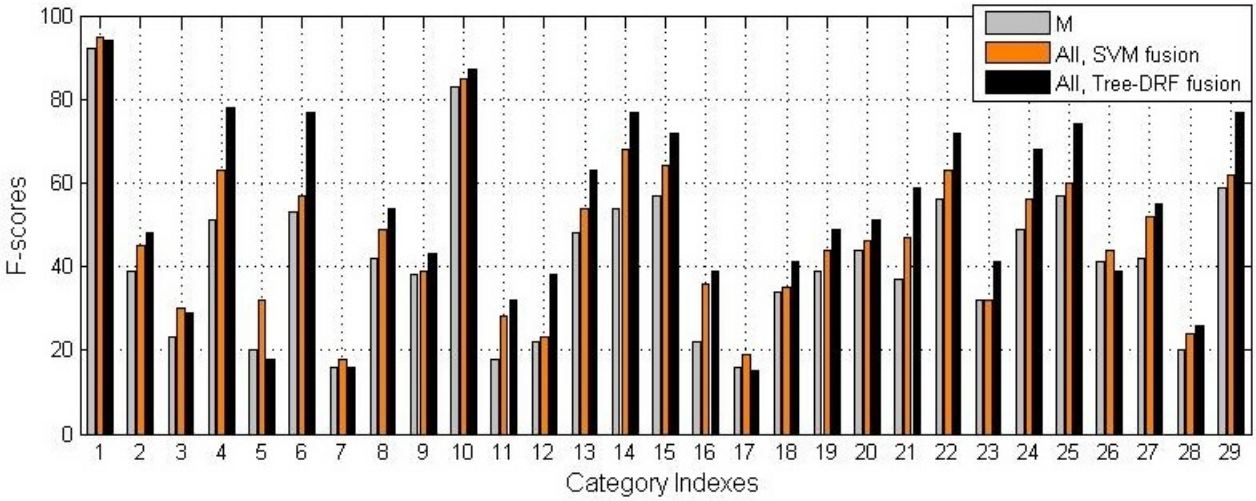

Figure 6. F-scores of 29 categories on manually-labeled data (M), all data with SVM fusion and all data with tree-DRF fusion. Tree-DRF performed better than the other two methods for most categories.

[20] G. Schindler, L. Zitnick, and M. Brown. Internet video category recognition. In *Proc. of CVPR Workshop on Internet Vision*, 2008.

[21] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *Proc. of ACM Workshop on Multimedia Information Retrieval*, 2006.

[22] C. G. M. Snoek, M. Worring, and A. W. M. Smeulders. Early versus late fusion in semantic video analysis. In *Proc. of ACM MM*, 2005.

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

[24] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive svms. In *Proc. of ACM MM*, 2007.

[25] S. Zanetti, L. Zelnik-Manor, and P. Perona. A walk through the web's video clips. In *Proc. of CVPR Workshop on Internet Vision*, 2008.

[26] H. Zhang, A. Kankanhalli, and S. W. Smoliar. Automatic partitioning of full-motion video. *Multimdedia Systems*, 1(1):10–28, 1993.

[27] X. Zhu. Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison, 2008.

[28] X. Zhu and Z. Ghahramani. Learning from labeled and unlabeled data with label propagation. Technical Report CMU-CALD-02-107, Carnegie Mellon University, 2002.