Improving Video Classification via YouTube Video Co-Watch Data

John R. Zhang*
Dept. Computer Science
Columbia University
New York, USA
jrzhang@cs.columbia.edu

Yang Song
Google, Inc.
1600 Amphitheatre Pkwy
Mountain View, USA
yangsong@google.com

Thomas Leung
Google, Inc.
1600 Ampitheatre Pkwy
Mountain View, USA
leungt@google.com

ABSTRACT

Classification of web-based videos is an important task with many applications in video search and ads targeting. However, collecting labeled data needed for classifier training may be prohibitively expensive. Semi-supervised learning provides a possible solution whereby inexpensive but noisy weakly-labeled data is used instead. In this paper, we explore an approach which exploits YouTube video co-watch data to improve the performance of a video taxonomic classification system. A graph is built whereby edges are created based on video co-watch relationships and weakly-labeled videos are selected for classifier training through local graph clustering. Evaluation is performed by comparing against classifiers trained using manually labeled web documents and videos. We find that data collected through the proposed approach can be used to train competitive classifiers versus the state of the art, particularly in the absence of expensive manually-labeled data.

Categories and Subject Descriptors

I.5.2 [**Pattern Recognition**]: Design Methodology

General Terms

Experimentation

Keywords

web videos, video co-watch, noisy data, video classification, semi-supervised learning

1. INTRODUCTION

Classification of videos is an increasingly prominent area of research, rising with the quantity of videos shared online through sites such as YouTube[1]. Its applications are of paramount importance to video search and website monetization through ads targeting. One challenge in building accurate video classifiers is the lack of manually labeled training data [4], which can be prohibitively expensive to collect. Semi-supervised learning techniques poses a solution, whereby inexpensively collected but potentially noisy weakly-labeled and unlabeled data are used instead.

In this paper, we propose and evaluate the efficacy of one approach to selecting weakly-labeled videos for classifier training based on combined viewer social information and text and content information from videos and web documents. Social information, revealed by viewers' co-watch habits, induces a *video co-watch graph* which is then filtered using a local graph clustering technique with the goal of reducing label noise in the final training set. For our application, we use a large-scale video taxonomic classification system which assigns one or more category labels to wild web videos [13]. We evaluate on videos collected from YouTube.

The video classification system assigns semantic categories to videos from a taxonomy. This taxonomy is organized as a tree 5 levels deep with 1037 nodes, with each node corresponding to a category. Some examples of top-level categories include *Arts & Entertainment*, *Adult*, *Autos & Vehicles*, *Beauty & Fitness*. These categories have children (i.e., subcategories) (a relationship denoted using a "/"), for example: *Arts & Entertainment / Comics & Animation*, *Arts & Entertainment / Movies*, *Autos & Vehicles / Vehicle Brands / Toyota* (a depth-3 category).

Our paper will be presented as follows. Section 2 will briefly discuss existing work in related areas. Section 3 will describe the large-scale video classification system and the design of its component classifiers for text and video. In Section 4 we present the sources of our data and how they are collected. Section 5 describes our approach for selecting weakly-labeled training data from the video co-watch graph through graph clustering. We use the collected data as training sets for experimentation in Section 6. We compare the performance of the models trained using the various sources of data, before concluding in Section 7.

2. RELATED WORK

Using images and videos collected from the web to train machine vision algorithms has become quite common [12, 9]. However, we aim to exploit more than simply the large quantity of data the web offers. Davison argues that most web pages are linked to others with related content [3]. We transfer this argument to the domain of online videos for our work here.

*Work done as an intern at Google while a student at Columbia University

[1]http://www.youtube.com

SBNMA'11, December 1, 2011, Scottsdale, Arizona, USA.

Mahajan et al. also exploits this hypothesis more directly by using the web graph to aid in classification of images [10]. In addition to image content features and text features (using text found in the web pages containing the images), label propagation through the web graph built using the hyperlinks in the encompassing web pages also contributes to classification. Their work focuses on the classification of adult content.

Wu et al. proposed the integration of social information for video classification in a similar setting [15]. The social and contextual information they use include related videos (as determined by YouTube) and user upload habits (they hypothesize that users upload videos largely belonging to the same category). Classification is done by combining scores from an SVM trained using videos' text metadata, and confidences computed using the sets of related videos and user's uploaded videos. Evaluation is performed using over 6,000 videos across 15 categories. While our work also uses related video data, the proposed approach exploits social information across viewers to handle noise in weakly labeled training sets and evaluate using a significantly larger number of categories and data set.

Other methods for handling noise in training sets have been studied in the field of semi-supervised learning [16]. Co-training is one approach whereby a feature set for a classifier is partitioned into two and data most confidently labeled by one is used as training data for the other in an iterative process, with the initial training done using labeled data [2, 7]. Co-training, however, requires that both feature set partitions are capable of training good classifiers. The semantic nature of our categories (e.g., *Cartoons* may contain both animated video and live action video of persons discussing cartoons) implies a poor separation of categories based on content features alone, resulting in an unsuccessful application of co-training. Therefore, we instead explored the use of the video co-watch graph to reduce label noise.

Leung et al. propose a multiple-instance learning approach for handling label noise in training sets for binary classifiers [8]. In this approach, positive samples are grouped into "bags", with each bag having at least one positive sample. By doing so, the effect of mislabeled training data on the final classifier is reduced. This approach can be applied to training sets selected using our proposed approach to further improve the performance of the final classifier.

3. SYSTEM OVERVIEW

A binary classifier is trained for each category in the taxonomy. We adopt the hierarchical one-against-all approach as in [13]. For a given category, the positive samples are those belonging to the category itself and its descendants, and the negative samples are the rest excluding those belonging to the ancestor categories. Classification decisions are based upon results from individual classifiers. Multiple labels can be assigned to one video.

Acquiring manually labeled web text documents is considerably cheaper than acquiring labeled video data and we assume that a good amount of manually labeled documents are available. To take advantage of this, our system includes two stages of training. The first stage is to train text-based classifiers from manually labeled web documents; and the second is to train classifiers using video data (text metadata and video content features). The same taxonomy is used for both stages.

3.1 Training Text-Based Classifiers From Labeled Web Documents

Text-based classifiers are learned from manually labeled web documents. A linear SVM is trained for each category in the taxonomy. Text features are represented by weighted text clusters, which are obtained from Noisy-Or Bayesian Networks [11].

These text-based classifiers serve two purposes. One is to act as one more layer of text feature extraction from video meta-data (more in Section 3.2.1); the other is to classify videos based on video meta-data. For categories without enough labeled (either manually or weakly) video data to train video-based classifiers (Section 3.2), these text classifiers are the only ones available to decide whether a video belongs to the category or not. These classifiers form the baselines in performance comparisons[2].

3.2 Training Classifiers From Videos

3.2.1 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 the same Noisy-Or Bayesian Networks as Section 3.1. In the second step, classifiers trained from Section 3.1 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 [13]. The second step exploits the knowledge embedded in the text-based classifiers learned from labeled web documents. The features obtained from the second step are more effective, especially when the number of training videos is small.

3.2.2 Video-Content-Based Feature Extraction

An assortment of video content-based visual and audio features are extracted, including histogram of local features, color histograms, edge features, face features, motion features, and audio features. For a more detailed description, please refer to [13] and references therein.

3.2.3 Classifier Training and Usage

The combined text and content features are used as candidate features to train Adaboost [5] classifiers. Decision stumps are used as weak classifiers.

4. VIDEO DATA SOURCES

Three sources are explored to gather video training data: the first is via expert human raters, the second is through some available (weak) classifiers to provide weakly labeled data; and the third is from a small seed set plus statistics on human social behavior (YouTube co-watch data). These three sources are detailed below. All of the videos are from YouTube.

4.1 Manually Labeled Videos

Human raters are asked to manually label a collection of videos for use as ground-truth data. Each of these raters has experience labeling web documents according to the same taxonomy and has demonstrated high inter-annotator consistency, therefore, they can be referred to as experts.

[2]We manually adjust the operating thresholds of these classifiers so that they work better on video metadata.

Raters are presented with videos to watch and asked to provide the categorie(s) (the deepest possible) which they feel are applicable. Because of their familiarity with the taxonomy, they are able to do this well despite the size of the taxonomy.

Manually labeled videos are expensive to collect (versus, say, similarly labeled web documents) since videos have a temporal span and raters must watch them (at least partly) before labeling. Over the course of a year, 10,528 videos were labeled in this way.

4.2 Videos From Weak Classifiers

Text classifiers described in Section 3.1 can be used as weak classifiers to generate (weakly) labeled video training data. The input to the text classifiers is the video metadata, such as title, description and keywords. Since the text classifiers are trained on web documents, their accuracy on videos is not necessarily very high. As a noise-reduction step, we only retain videos which result in a high confidence score from the classifiers. However, a side effect of this could be a biased set of selected videos.

4.3 Related Videos

Anonymized user sessions were analyzed to identify co-watched videos. That is, if the number of users who watched the same two videos in one session exceeds some threshold, then the two videos are considered to be co-watched. This data reveals the underlying social relationships between viewers and has been used for other applications such as community detection [6]. Intuitively, co-watched videos are likely to belong to the same categories, since users tend to watch similar videos in the same session. Starting from a small set of *seeds* (i.e., manually labeled) videos, we use co-watched videos to expand our training set. Section 5 describes how these co-watched videos (one or more hops from the seeds) are effectively gathered and used.

5. LOCAL GRAPH CLUSTERING OF RELATED VIDEOS

The co-watched videos provided by YouTube (Section 4.3) may be highly noisy. That is, our hypothesis that users tend to watch similar videos in the same session is not always true. We discuss a method here which we to reduce mislabelings.

Let the collection of online videos be represented as vertices in a directed graph $G = (V, E)$, where an edge $(u, v) \in E$ exists if videos u, v are co-watched (as defined in Section 4.3). Suppose we are also provided with a set of seed (manually labeled) videos S, and we wish to augment this set with weakly labeled videos to produce S'. Simply taking the one-hop co-watched videos $S' = \{c|(c, s) \in E, s \in S, c \notin S\} \cup S$ is one option. Wang et al. did this to augment their training data but added the restriction that videos must be co-watched more than 100 times to be included in their training set [14]. No further details were given regarding the effect of the added videos, as noise was expected to be handled by a different aspect of their work.

We propose an alternative which allows for the discovery of a greater number of related videos in a more principled way, by including videos from multiple hops away as well. Naturally, with multiple hops, the number of unrelated videos found would be even larger. To counteract this, one can argue that a video that is co-watched with only one

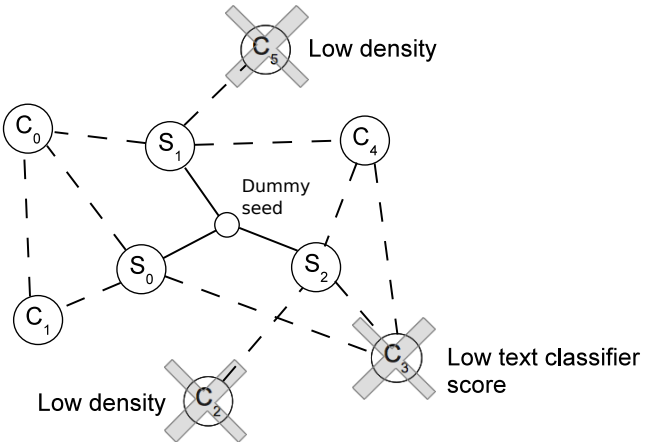

Figure 1: Overview of the graph clustering for each category. The seed videos s_i are inserted as co-watched videos of a dummy seed with infinite edge weights. At each iteration, the co-watched videos c_j for each of the existing videos in the cluster are found and added to the cluster. After every constant k iterations, the cluster is pruned: vertices with low degrees and those with a low likelihood of belonging to the category (as determined by the text classifier) are removed.

video in the seed set S has a higher likelihood of being unrelated. However, if that same video has been co-watched with multiple seed videos, then the likelihood of it being related increases. We can discover these videos through graph clustering.

Since a large number of videos (on the order of millions) need to be exploited to discover significant co-watch relationships, a local graph clustering method is used. We use Andersen's dense partitioning algorithm [1], which, when given input graph G and seed vertex v, returns a dense subgraph around v. Although the algorithm is designed for undirected bipartite graphs, the authors state that it can be adapted for arbitrary directed graphs as well, in which case the density for a desired cluster S' can be expressed as the following equation:

$$d(S') = \frac{e(S', S')}{|S'|} \tag{1}$$

where $e(S', S')$ is the sum of the weights of edges between vertices in S'.

Andersen's algorithm works iteratively: starting from the seed vertices (a dummy vertex connected to the actual seed vertices we want can be used to get around the requirement of a single seed vertex), directly connected vertices to the existing cluster are added, and at regular intervals the cluster is pruned. When adding a co-watched video to the cluster, a weight is assigned to the edges connected to it based on its rank in the list of co-watched videos with its neighboring vertices. These weights are propagated from the seed videos so that videos further from the seeds would have decreasing weight. The weight associated with each node is then simply the sum of the weights of the incident edges. During pruning, vertices with low weights are removed. We augment this pruning step by also classifying added videos using the text classifier (Section 3) and only retaining those that produce a

sufficient score. Empirically, this step greatly increases the accuracy of the clusters produced. An overview of the graph clustering is illustrated in Figure 1.

6. EXPERIMENTS

Different approaches of generating training data are evaluated by comparing the performance (as measured by precision, recall and F-score) of the resultant classifiers. We perform 5-fold (cross) validation on 10,528 manually labeled videos. Note that this naturally biases classification results to favor those models trained using manually labeled data.

We also have at our disposal a large corpus of web documents, which are manually labeled according to the same taxonomy. We use them to train all our text classifiers. As it is not the focus of this work, a detailed discussion of this data is omitted.

Unlabeled video data is collected from a pool of approximately 24 million videos randomly selected from YouTube. The related videos and the videos selected by the text classifiers are a strict subset of this. Note that not every single one of these 24 million videos are necessarily used.

6.1 Comparison

We want to compare classifiers trained on automatically (i.e., weakly) labeled videos versus baseline classifiers training using manually labeled videos (when a sufficient number of training videos exist) or direct application of the text classifiers. The weakly labeled training sets are as follows:

- **Videos selected by text classifier**. Using classifiers described in Section 3.1, labels are assigned to a randomly selected collection of videos. From each category we take 250 videos for which the text classifier has confidently (i.e., assigned a score greater than a threshold of 0.99 out of 1.0) assigned to that category.
- **Videos selected based on co-watch relationships (related videos)**. We apply the methods described in Section 4.3 to augment video training data. Some partitions of manually labeled video data are used as seeds. In our experiments, we cap the seed sizes at 100 videos. We experiment with 2 different augmented training set sizes: 150 and 250. These videos are called "related videos" in future descriptions and figure captions. We will discuss the effect of different training set and seed set sizes in Section 6.2.

Since the end goal is to improve the video classification system on the whole, we can take advantage of the fact that it is comprised of binary classifiers for each category. That is, we can mix models by taking the best performing model trained using these data sets and combine them in the final application. We denote this as the "Max" set of classifiers in the figures. The creation of this set of best models could be done automatically through a validation step.

The baselines are as follows:

- **Classifiers trained on manually labeled videos**. These classifiers are trained for categories with at least 100 manually labeled videos. Classifiers for only 68 categories were produced, presenting a clear need for semi-supervised techniques.
- **Direct classification by text classifier**. Test videos are classified based on their text features (i.e., metadata) alone. When there are insufficient manually labeled training videos, this is the state-of-the-art [13].

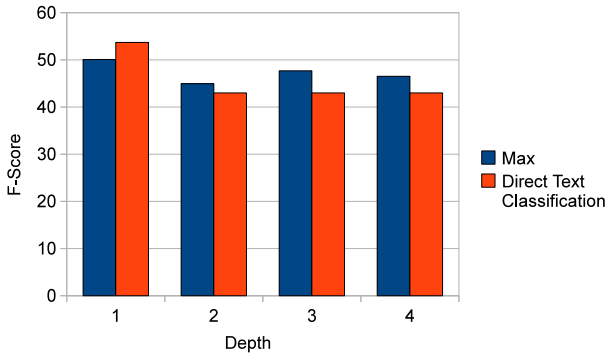

Figure 5: Comparison of average F-score across depths. Blue bars show the average of "max" performance from weakly-labeled data; red bars show the average performance for "direct text classification".

Figure 2 shows the performance comparison of models trained from different sources of weakly-labeled data: related videos and videos selected by text classifiers. For the training set collected from related videos, we examine both training set sizes of 150 and 250 videos. Some analysis of the performance is given in Section 6.2.

For categories with sufficient manually labeled video data, classifiers with "Max" performance from weakly-labeled data are compared against those trained using manually labeled data. The results are shown in Figure 3. For a significant number of categories, the performance of the classifiers trained using weakly-labeled data are competitive with those trained using manually labeled data.

For categories without sufficient manually labeled video data, classifiers with "Max" performance from weakly-labeled data are compared against the baseline of direct text classification. The results are shown in Figure 4. The categories shown in the figure have between 75 to 100 manually labeled videos. The lower bound of 75 is semi-arbitrarily selected so that a number of models could be displayed here. For categories with 10 to 100 manually labeled data, automatically collected videos trained superior models in 92 categories, and inferior models in 88 categories.

We can also examine the performance of classifiers according to depth instead of category, as shown in Figure 5. The performance of the classifiers at each depth are averaged in this figure. "Max" achieves better performance at most depth levels.

6.2 Effect of Co-Watched Video Cluster and Seed Sizes on Final Classifier Performance

We include here a brief discussion of the effects of different training set sizes and seed set size for co-watched video clustering on the performance of the trained classifier.

Given seed videos, the clustering method allows us to expand on them to various sizes. We experiment with training set sizes of 150 and 250 videos. Since the selection of these videos also depends on the weights assigned to them, which in turn are a function of their rank in the list of related videos, the number of videos in the training set affect the amount of noise in the set. That is, the larger training

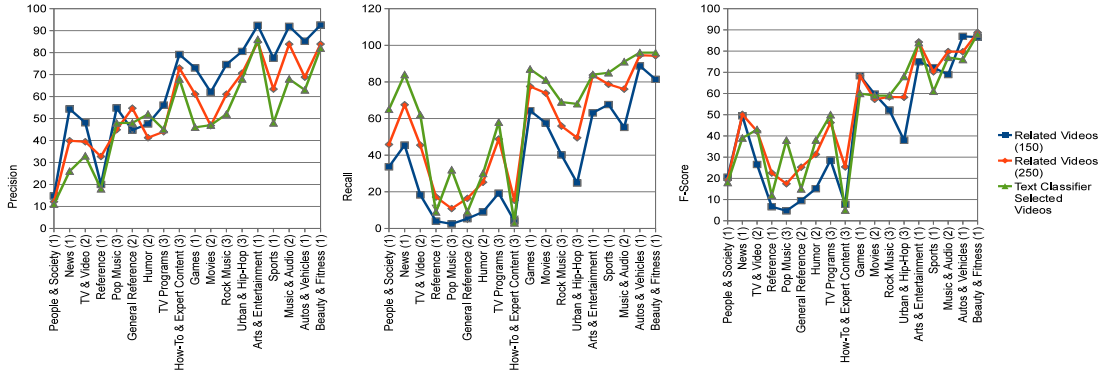

Figure 2: Comparison of classification performance of models trained using different sources of weakly-labeled videos: related videos with augmented training set size 150, related videos with augmented training set size 250, and videos selected by text classifiers. The categories are shown with their depths in parentheses.

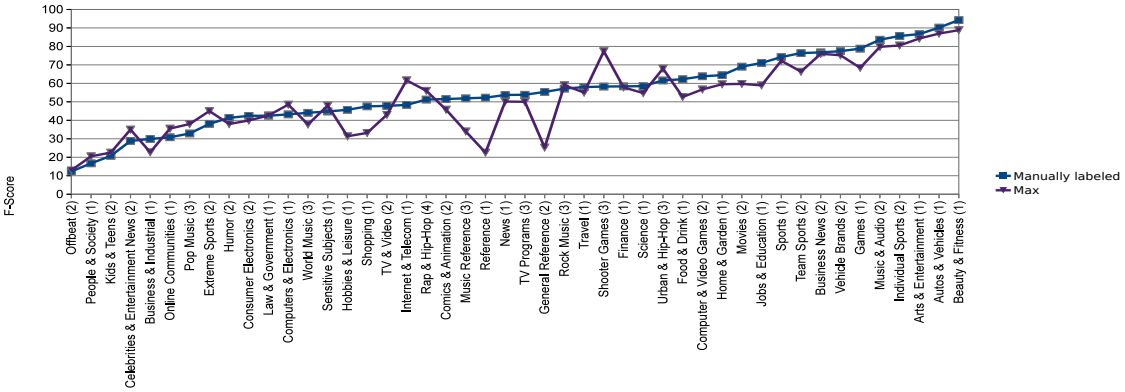

Figure 3: Performance comparison for categories with sufficient manually labeled data: "max" performance from weakly-labeled data vs. performance from manually labeled data.

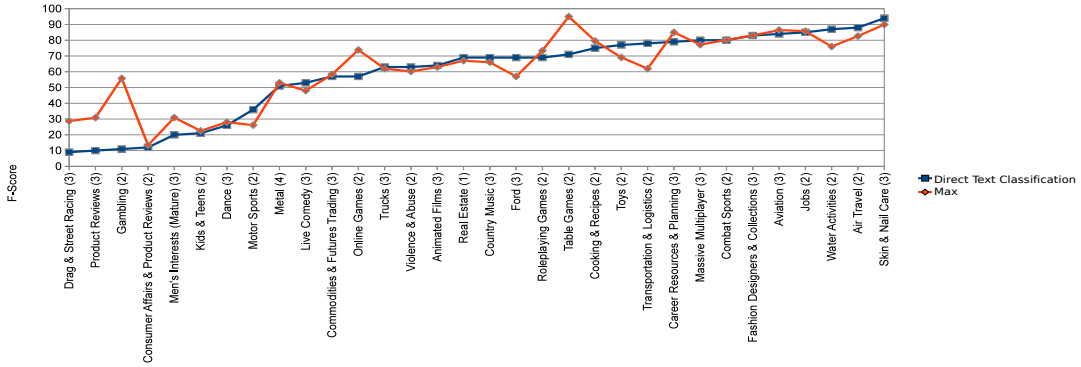

Figure 4: Performance comparison for some categories without sufficient manually labeled data: "max" performance from weakly-labeled data vs. performance from "direct text classification".

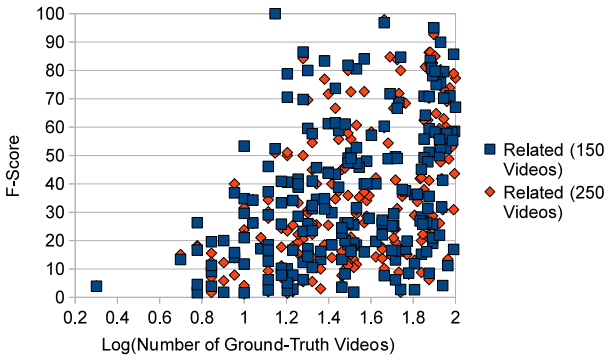

Figure 6: Correlation between the number of ground-truth videos available (i.e. seeds) and F-score of the classifier trained using related videos.

set would have more mislabeled videos. This is reflected in the final comparison of the classification performance of the models trained using these sets, as shown in Figure 2, where the smaller training set is capable of producing higher precision but lower recall due to the decrease in label noise and the decrease in data diversity.

The correlation between the size of the seed set with the final classifier performance is depicted in Figure 6 (graph is truncated to show only categories with up to 100 seed videos). There is a positive correlation between the classifier performance (F-score) and the seed size (number of ground-truth videos).

7. CONCLUSION

We have presented a method to use YouTube video co-watch data to generate training data and to improve the performance of a video taxonomic classification system. A graph clustering method is proposed to select video training data using co-watch statistics. Classifier models are trained using video data from different sources: videos from co-watch data, video from some pre-trained (weak) classifiers, and manually labeled video data. Extensive experiments are performed to compare these models. Models from our proposed methods have shown superior performance to the existing baseline models when there are not enough manually labeled video data.

Future work to improve the performance of the classifiers could focus on exploring methods to reduce the label noise in the automatically collected training videos.

8. REFERENCES

[1] R. Andersen. A local algorithm for finding dense subgraphs. In *Proc. 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1003–1009, 2008.

[2] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. 11th Annual Conference on Computational Learning Theory*, July 1998.

[3] B. D. Davison. Topical locality in the web. In *Proc. 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 272–279, 2004.

[4] R. Fergus, H. Bernal, Y. Weiss, and A. Torralba. Semantic label sharing for learning with many categories. In *Proc. European Conference on Computer Vision*, September 2010.

[5] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, pages 119–139, 1997.

[6] U. Gargi, W. Lu, V. Mirrokni, and S. Yoon. Large-scale community detection on youtube for topic discovery and exploration. In *To appear, International AAAI Conference on Weblogs and Social Media*, 2011.

[7] S. Goldman and Y. Zhou. Enhancing supervised learning with unlabeled data. In *Proc. 17th International Conference on Machine Learning*, pages 327–334, 2000.

[8] T. Leung, Y. Song, and J. R. Zhang. Handling label noise in video classification via multiple instance learning. In *To appear, ICCV*, 2011.

[9] L.-J. Li, R. Socher, and L. Fei-Fei. Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009.

[10] D. Mahajan and M. Slaney. Image classification using the web graph. In *Proc. ACM Multimedia*, 2010.

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

[12] J. C. Niebles, B. Han, A. Ferencz, and L. Fei-Fei. Extracting moving people from internet videos. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, pages 527–540, 2008.

[13] Y. Song, M. Zhao, J. Yagnik, and X. Wu. Taxonomic classification for web-based videos. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2010.

[14] Z. Wang, M. Zhao, Y. Song, S. Kumar, and B. Li. Youtubecat: Learning to categorize wild web videos. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2010.

[15] X. Wu, W.-L. Zhao, and C.-W. Ngo. Towards google challenge: Combining contextual and social information for web video categorization. In *ACM Multimedia*, 2009.

[16] X. Zhu. Semi-supervised learning literature survey. In *Tech Report*. University of Wisconsin—Madison, July 2008.