Semi-supervised Cross Feature Learning for Semantic Concept Detection in Videos

Rong Yan

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, 15213

Milind Naphade *

IBM TJ Watson Research Center
19 Skyline Drive
Hawthorne, NY, 10532

Abstract

For large scale automatic semantic video characterization, it is necessary to learn and model a large number of semantic concepts. But a major obstacle to this is the insufficiency of labeled training samples. Multi-view semi-supervised learning algorithms such as co-training may help by incorporating a large amount of unlabeled data. However, one of their assumptions requiring that each view be sufficient for learning is usually violated in semantic concept detection. In this paper, we propose a novel multi-view semi-supervised learning algorithm called semi-supervised cross feature learning(SCFL). The proposed algorithm has two advantages over co-training. First, SCFL can theoretically guarantee its performance not being significantly degraded even when the assumption of view sufficiency fails. Also, SCFL can also handle additional views of unlabeled data even when these views are absent from the training data. As demonstrated in the TRECVID'03 semantic concept extraction task, the proposed SCFL algorithm not only significantly outperforms the conventional co-training algorithms, but also comes close to achieving the performance when the unlabeled set were to be manually annotated and used for training along with the labeled data set.

1 Introduction

Increasingly, the detection of a large number of semantic concepts is being seen as an intermediate step in enabling semantic video search and retrieval[1, 2, 3]. These semantic concepts cover a wide range of topics that can be roughly categorized as objects, sites, events, and specific personalities and named entities. The main idea of semantic concept detection is to treat it as a statistical learning problem. For each video shot, the associated concepts can be detected using multiple unimodal classifiers or multimodal classifiers [4, 5] based on visual, audio and text/speech features. Using a large annotated corpus, these concepts can be learnt if sufficient number of training samples exist. Unfortunately, annotation is a labor-intensive process and the number of labeled video samples is usually not enough for most semantic concepts. Typically annotating 1 hour of video divided into shots, with a lexicon of 100 semantic concepts can take anywhere between 8 to 15 hours [6]. The problem is further worsened for a large number of concepts which appear infrequently.

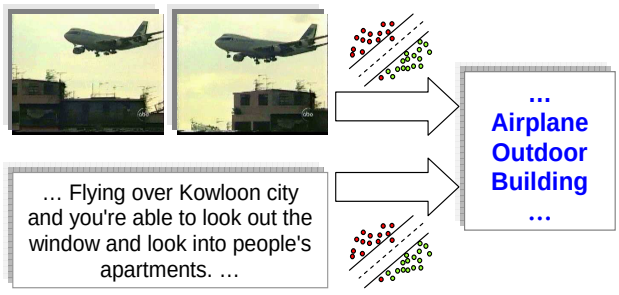


Figure 1: Illustration of detecting semantic concepts from video sequences. Each video shot is associated with multimodal information including both text/speech transcript and visual frames. The semantic concepts can be detected by combining the outputs of multiple unimodal classifiers.

One way to deal with insufficient labeled data is to apply the semi-supervised learning algorithms which attempt to leverage a large amount of unlabeled data set to boost the classification accuracy. The multiple modalities of the video stream further suggest considering the multi-view setting which explicitly split the feature space into multiple subsets, or views. Combining semi-supervised learning and multi-view setting offers a more powerful way to leverage unlabeled data. Co-training[7] is one of the most well-known multi-view semi-supervised learning al-

*This material is based upon work funded in part by the U.S. Government. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the U.S. Government.

gorithms. The co-training algorithm starts with two initial classifiers learned from each view separately. Both classifiers are then incrementally updated in every iteration using an augmented labeled set, which includes additional unlabeled samples with the highest classification confidence in each view. Co-EM[8] can be viewed as a probabilistic version of co-training, which requires each classifier to provide class probability estimation for all unlabeled data. Collins and Singer[9] introduced the CoBoost algorithm which attempts to minimize the disagreement on the unlabeled data between classifiers of different views. This class of co-training type algorithms has been successfully applied to a variety of real-world domains, from natural language processing[10], web page classification[7], information extraction[9] to visual detection[11]. However they have not been successfully applied in the domain of video concept detection, which has been considered a potentially applicable domain by Blum et al[7]. After examining the real-world video data, we realized that one reason for the failure of the co-training algorithms in this domain is the violation of the underlying assumptions which requires that each view be sufficient for learning the target concepts. Therefore it is possible for the co-training type algorithms to greatly deteriorate the learning performance when a large unlabeled data set is used.

In this paper, we propose a novel multi-view semi-supervised learning algorithm called semi-supervised cross feature learning(SCFL). It intends to relax the view sufficiency assumption in co-training. Unlike co-training which updates each classifier by incorporating the selected unlabeled data to augment the labeled set, SCFL learns separate classifiers from the selected unlabeled data and combines them with the classifiers learned from noise-free labeled data. The proposed approach provides two advantages over the co-training algorithm. First, SCFL can theoretically prevent its performance from being significantly degraded even when the assumption of view sufficiency fails. Also it allows additional views of unlabeled data to be incorporated even when these views are absent from the training data. In the following sections, we first briefly review the co-training algorithm and then describe the details of the proposed SCFL algorithm. Its effectiveness for detecting video concepts is demonstrated using a large video archive from TRECVID'03 concept extraction task[1].

2 The Co-Training Algorithm

The co-training algorithm belongs to a class of algorithms that combine semi-supervised learning and multi-view learning into one unified framework. Formally, the goal for co-training is to learn a classifier $f(x)$ using both labeled data $L : \{(x_1, y_1), ..., (x_n, y_n)\}$, and unlabeled data $U : \{x'_1, ..., x'_m\}$. The feature space can be split into two

---

Input Two views V_1 and V_2, labeled data L including training data L_t and validation data L_v, unlabeled data U, the number of iterations T

Co-Training

1. Create the classifier f_1^0 and f_2^0 using L_t on V_1 and V_2
2. For $t = 1, 2, ...T$
   (a) Remove n_p examples with largest $f_i^{t-1}(x')$ and n_n examples with smallest $f_i^{t-1}(x')$ from the unlabeled set U, $i = 1, 2$
   (b) label the selected examples according to f_1^{t-1} and f_2^{t-1} and add to the training set L_t
   (c) Create the classifier f_1^t and f_2^t using L_t on V_1 and V_2
3. Combine $f^T = w_1 f_1^T + w_2 f_2^T$ using L_v

Figure 2: The co-training algorithm

disjoint views V_1 and V_2, and thus each labeled example (x_i, y_i) can be decomposed into (x_{i1}, x_{i2}, y_i) where x_{i1} and x_{i2} are the features over the views V_1 and V_2 respectively. The classifier learned from view V_j is denoted as $f_j(x)$.

The approach of co-training is to incrementally update the classifiers of multiple views which allows the redundant information across views to improve the learning performance. For each view V_j, the classifier $f_j(x)$ is first initialized by learning a few labeled examples L_t. At each iteration, the algorithm will select a batch of unlabeled data from the unlabeled set U to incorporate into the pool of labeled data L_t. Typically these additional unlabeled data are selected as those with the highest prediction confidence for each view and assigned with the corresponding labels. Each classifier $f_j(x)$ is then updated from the augmented labeled data set. This process is iterated until T iterations. Finally, weighted linear combination of the output classifiers $f_j(x)$ gives a single-view classifier $f(x)$, where the weights w_i are obtained from a validation set L_v. Note that we are not strictly following the conventional co-training setting which suggests w_i are equal, because in our application not all the views are sufficient enough or equally relevant to capture the underlying concept and hence they should not have the same weights. The validation data L_v, of which the number is much smaller than the training data L_t, is used to determine the combination weights w_i.

The intuition of co-training is that the two classifiers can provide each other with new labeled data which might be as informative as some random noisy labeled examples. Based on the analysis of Blum et al[7], the success of co-training requires two underlying assumptions. The first one is the

conditional independence which means the views V_j should be conditionally independent of each other in order to provide useful information. The second one is the view sufficiency which means each view should be itself sufficient for learning the target concept and thus all examples are labeled identically in each view.

The assumption of conditional independence might be reasonable in the task of video concept detection because the text modality can be viewed as an independent source of the visual modality. But the assumption of view sufficiency will not generally hold in practice. For example, when the feature of color histogram is used to learn the video concept "airplane", of two video frames which have the same color histogram, one might contain an airplane but the other might contain an eagle. Therefore the view from color distribution will not be sufficient to learn the underlying concept "airplane". Since most concepts are capturing the semantic meaning, it is difficult for the low-level visual features alone to sufficiently represent the concepts. In this case, co-training will usually exhibit poor performance as shown in our experiments. In the domain of natural language processing, Pierce et al[10] also observed the similar degradation of co-training algorithm if the labeled data introduced by the other view is not accurate enough.

To mitigate this problem, several algorithms such as corrected co-training[10] and co-testing[12] were proposed. The major idea of these algorithms is to rely on a human annotator to review and provide the correct labels for selected unlabeled data. However when dealing with unlabeled data, we usually assume that no further manual annotation is possible. Ideally, we should guarantee that the unlabeled sets will at worst result in no significant performance degradation and at best improve performance over the use of the labeled data sets alone.

3 Semi-Supervised Cross Feature Learning

In this section, we propose a new multi-view semi-supervised learning algorithm called semi-supervised cross feature learning (SCFL), which attempts to alleviate the problems of co-training when some views are not sufficient to learn target concepts by themselves. We first describe the details of the SCFL algorithms on two views, followed by a theoretical analysis showing its advantage. Finally, an extended SCFL algorithm in an attempt to handle multiple views is presented.

3.1. Algorithm

As mentioned before, the main concern for applying the co-training algorithm is that when view sufficiency assumption fails, the additional training data with random classification noise are likely to corrupt the initial classifiers. With more noisy training data introduced, co-training tends to iteratively produce worse performance. Alternately we prefer some other approaches that can automatically detect the deterioration of existing classifiers and avoid combining the poor classifiers into the final output.

Input Two views V_1 and V_2, initial classifiers f_1^0 and f_2^0 on V_1 and V_2, additional training data $L'_t = \emptyset$, validation data L_v, unlabeled data U, the number of iterations T

Semi-Supervised Cross Feature Learning

1. For $t = 1, 2, ...T$
   (a) Remove n_p examples with largest $f_i^0(x')$ and n_n examples with smallest $f_i^0(x')$ from the unlabeled set U, $i = 1, 2$
   (b) label the selected examples according to f_1^0 and f_2^0 and add to L'_t
2. Create the classifier f'_1 and f'_2 using L'_t on V_1 and V_2
3. Combine $f^T = \sum_{i=1}^{2} w_i^0 f_i^0 + \sum_{i=1}^{2} w'_i f'_i$ using L_v

Figure 3: The semi-supervised cross feature learning

To improve upon co-training, we propose the semi-supervised cross feature learning(SCFL) algorithm, of which the rationale is to separate the prediction of initial classifiers and the prediction using additional unlabeled data into different ensembles. As shown in Figure 3, the SCFL algorithm starts with the initial classifiers $f_i^0(x)$ which can be trained on the labeled set alone from each view V_i. In the first step, we collect a batch of unlabeled data from U and label them automatically using the initial classifiers. [1] Then two separate classifiers from each view will be learned solely from the unlabeled data. Finally all four classifiers f_1^0, f_2^0, f'_1, f'_2 will be weighted combined based on the validation data V. With aid of the validation set L_v, the SCFL algorithm detects how much benefit can be achieved from the unlabeled data without hurting the performance of initial classifiers. If the predictions from unlabeled data is too noisy to be useful, the worst scenario will be setting their weights w'_i to 0 which naturally backs off to the combination of two initial classifiers. More detailed analysis for our approach follows in Section 3.2.

[1] Note that although in this step we explicitly iterate the sampling procedure to indicate the connections between co-training and SCFL, the sampling can be computed without iterations by directly sampling a batch of $n_p * T$ positive examples and $n_n * T$ negative examples based on f^0.

3.2. Analysis

In order to theoretically understand why SCFL can outperform co-training when the assumption of view sufficiency fails, let us focus on the final step of the SCFL algorithm. This step is crucial for controlling the learning quality which prevents the useless predictions being combined into the final output. Formally, it attempts to learn a linear meta-classifier f on top of all the available classifiers f_i and f'_i using the validation data L_v. Given the labeled data $(x_1, y_1), ..., (x_n, y_n)$ from V, the inputs for the meta-classifier f are the vectors of classifier outputs and labels $\{(f_0^0(x_i), f_1^0(x_i), f'_0(x_i), f'_1(x_i)), y_i\}, 1 \leq i \leq n$. Let $\mathcal{F}$ denote the collection of classifiers $f : \mathbf{R} \rightarrow \{0, 1\}$ and h denote its VC dimension. We can define

$$f_{emp} = \arg\min_{f \in \mathcal{F}} R_{emp}(f)$$

where

$$R_{emp}(f) = \frac{1}{n} \sum_{i=1}^{n} I(f(x_i) \neq y_i)$$

is the empirical risk of f. Thus f_{emp} is selected by empirical risk minimization on $\mathcal{F}$. Similarly, we can define

$$f_0 = \arg\min_{f \in \mathcal{F}} R(f),$$

where $R(f)$ is the generalized risk of f. Then we have the following risk bounds of f_{emp}[13],

Theorem 1 *The following risk bound holds in terms of the VC dimension h and number of training data n with probability $1 - 2\eta$,*

$$R(f_{emp}) \leq R(f_0) + Bound(h, n, \eta)$$

where

$$Bound(h, n, \eta) = \sqrt{\frac{h(\ln(2n/h) + 1) - \ln(\eta/4)}{n}} - \sqrt{-\frac{\ln \eta}{2n}}.$$

For the proposed approach, the minimal risk $R(f_0)$ is always lower than $R(f_1^0, f_2^0)$ which is the minimal risk of combining two initial classifiers alone. Since $R(f_1^0, f_2^0)$ does not change after unlabeled data are incorporated, based on Theorem 1 the generalized risk of the trained model $R(f_{emp})$ will be bounded by $R(f_1^0, f_2^0)$ and $Bound(h, n, \eta)$ which is generally small with VC dimension $h = 5$. This guarantees the performance of the meta-classifier f_{emp} will be bounded even after lots of unlabeled data are introduced.

However, the co-training algorithm does not have such guarantee any more, because both f_1^T and f_2^T can be deteriorated when a large number of unlabeled data are included and thus the minimal risk $R(f_0)$, i.e. the minimal risk of combining f_1^T and f_2^T, is not necessarily lower than the risk of $R(f_1^0, f_2^0)$. Therefore, if some views are not sufficient to learn the target concepts, the co-training algorithm will probably suffer from a considerable performance degradation.

Readers might come up with one question: Why not directly leverage the annotated validation set L_v to learn the target concept, since the generalized risk is bounded? Actually if the concept is learned directly from L_v, the VC dimension will become much larger than $h = 5$ while the number of L_v is typically much smaller than training data. It will lead to a unreasonable risk bound and produce poor generalized results. Using the L_v for weight learning turns out to be a better choice.

Input Initial classifiers f_i^0 on multiple views $V_i (1 \leq i \leq n)$, additional training data $L'_t = \emptyset$, unlabeled data U on view $V'_j (1 \leq j \leq m)$, validation data L_v on views $\{V_i\} \cup \{V'_j\}$, the number of iterations T

Extension to multiple views

1. For $t = 1, 2, ...T$, similar to Step 1 in Figure 3 except n views are considered
2. Create classifiers f'_j using L'_t on view $V'_j, 1 \leq j \leq m$
3. Combine $f^T = \sum_{i=1}^{n} w_i f_i^0 + \sum_{j=1}^{m} w'_j f'_j$ using L_v

---

Figure 4: Extend the SCFL algorithm to multiple views

3.3. Extension to multiple views

Introducing more views into learning is another way to improve the classification performance besides introducing more training data. However, in most previous work the co-training algorithm only considers using two views instead of multiple views. Especially if the unlabeled data set U contains more views than the training data set L_t, the additional views will not be able to handled by the co-training algorithm. For example, if the texture features are available for U but not available for L_t, co-training will fail to utilize the texture features because they are absent in the training data, although this extra piece of information will be helpful to detect the concepts.

In contrast, a multi-view extension of the SCFL algorithm is able to learn on multiple views simultaneously and also utilize the additional views from the unlabeled data set. The details of the algorithms are shown in Figure 4. First, more than two initial classifiers from different views can be taken as the inputs. In step 1, a similar sampling strategy is

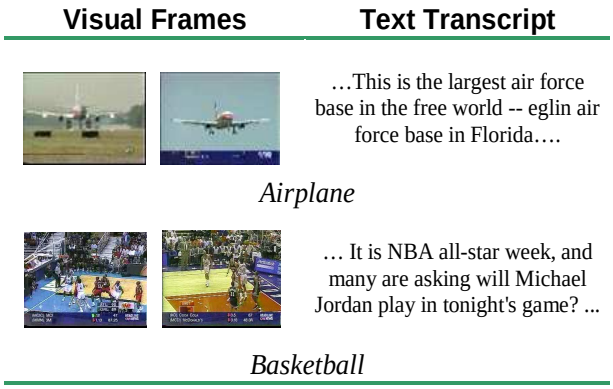


Figure 5: Visual and text labeled examples for the video concepts of "airplane" and "basketball"

applied to select the unlabeled data to label. Then the additional training set L'_t is used to learn multiple classifiers f'_j on views V'_j. Note that the number of views of unlabeled data V'_j can be more than that of the original views V_i, which means SCFL can handle the extra views from the unlabeled data. Again, all of these classifiers' predictions will be linearly combined using the validation data L_v. This multi-view extension of SCFL gives us more flexibility to improve the classification performance.

4 Experiments

4.1. Experimental Setting

To evaluate the performance of the proposed SCFL algorithm, we follow the guideline of TRECVID03 semantic concept extraction task[1] to design our experiments. The goal of the concept extraction task is to detect the presence or absence of a video concept in the reference video shots over a 65 hour news video corpus. Eleven concepts are selected for the task, i.e. *Airplane, Basketball, Bill Clinton, Beach, Boat, Animal, Hockey, Mountain, Rock, Tree and Water Body*. These concepts cover a broad range of interesting topics in news video and they could be detected from the low-level features with reasonable accuracy.

We construct our data set by partitioning the news video collection into four data sets, i.e. a training set, a validation set, a testing set and an unlabeled set, where each video shot is associated with the truth annotations over every concept[1, 6]. In the training set, we pick all the positive examples from the training set and down-sample the negative examples to keep the ratio between positive and negative examples to be 1:5. This ratio is chosen so as to provide a reasonable trade-off between the performance and the running time. Validation data are selected in a similar fashion from the validation set. On average for each concept we collect around 1800 shots for training and 300 shots for validation. In the following experiments, all of the unlabeled data and testing data are used without reference to their true labels, which includes 5037 shots and 9852 shots respectively.

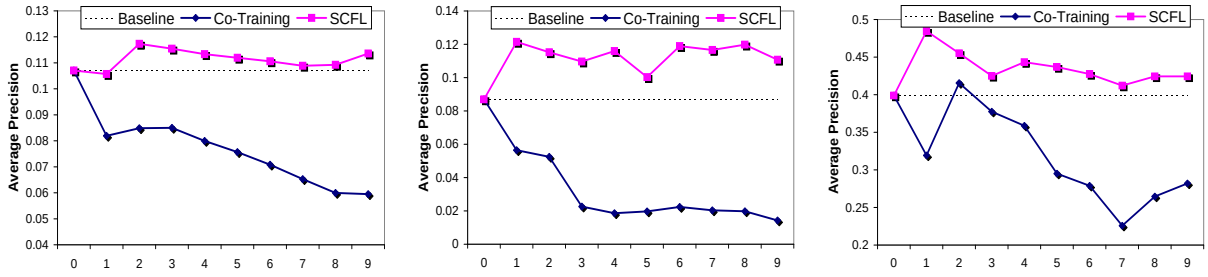
For each video shot, we extract two types of low-level features, i.e. 146 dimensional color correlogram feature vector in HSV color space and binary word presence features of automatic speech transcripts/closed caption[2]. In addition, the co-occurrence texture features [2] with 96 dimensions are extracted for the multi-view SCFL algorithm. Figure 5 shows some labeled video frames and text descriptions associated with concepts "airplane" and "basketball". The choice of color correlogram as image features are mainly determined by the characteristics of the TRECVID dataset. Since it is the largest available annotated video corpus so far, we have to choose image features that are fully scalable and universally applicable to a wide variety of concepts and a large collection of data. Previous experiments revealed that despite its simplicity, color correlogram have been one of the most effective features for this data set [2].

Because the number of positive data is usually much less than negative data in our task, the classification accuracy is not a preferred performance measure. Alternatively, NIST defines non-interpolated average precision over a set of retrieved shot as a measure of retrieval effectiveness. Let R be the number of true relevant documents in a set of size S. At any given index j, let R_j be the number of relevant documents in the top j documents. Let $I_j = 1$ if the j^{th} document is relevant and 0 otherwise. Assuming $R < S$, the non-interpolated average precision **(AP)** is then defined as $\frac{1}{R}\sum_{j=1}^{S}\frac{R_j}{j} * I_j$. Mean average precision(MAP) is the average of average precisions over all concepts.

4.2. Performance Evaluation

To provide a fair comparison, we use the same set of training data to produce the initial classifiers for all of the algorithms. SVM^{Light}[14] is adopted as the underlying classifier where the linear kernel is applied for text features and the RBF kernel for visual features. Cross validation is used to decide the learning parameters and the cost factor that achieve the best average precision on the training data. In the final step, we apply the Powell's direction set methods[15] to learn the combination weights which maximize the average precision on the validation set.

We design several multi-view learning algorithms as follows. The baseline algorithm (**Baseline**) simply combines the initial classifiers with labeled data only. Unlabeled data are utilized in both the co-training(**CoTrain**) and SCFL(**SCFL**) algorithm. They run up to the 9 iterations. In each iteration, we select additional unlabeled data as much as 10% of the training data. Therefore at the end of the learning process, the number of unlabeled data is about the same as the number of training data. Moreover, to examine the maximal benefit that can be gained from unlabeled data,



(a) Airplane (b) Boat (c) Hockey

Figure 6: The learning curves of three selected video concepts including (a) Airplane, (b) Boat and (c) Hockey. Three algorithms are compared: baseline, co-training and SCFL. They have the same performance in the first iteration when none of the unlabeled data is added. Each curve plots average precision against the number of iteration from 0 to 9. A fixed number of unlabeled data is added per iteration.

we report the result based on combining the classifiers with both the labeled data and the unlabeled data with their true labels(**FullLabel**). Finally, in order to show the advantages of incorporating additional views from unlabeled data, we also examine the SCFL algorithm with multi-view extension(**SCFL-M**). It uses extra co-occurrence texture features of the unlabeled data, but it keeps the same two initial classifiers as before and thus no initial classifiers on texture features are provided.

Figure 6 depicts the learning curves on three of our concepts, i.e. Airplane, Boat and Hockey, which represents the general learning patterns over various algorithms. Each subgraph includes the curves of three learning algorithms, i.e. **Baseline**, **CoTrain** and **SCFL**. We can observe that co-training tends to produce lower average precision with more unlabeled data introduced. For the concept "hockey" of which the baseline performance is the highest, co-training can outperform baseline when adding a small amount of unlabeled data, but when too many unlabeled data are incorporated the performance still degrades akin to other concepts. In contrast to co-training, SCFL can almost always provide better performance than baseline for each iteration, even for the concepts which cannot be detected very accurately. This observation corroborates the robustness of SCFL in various learning scenarios. However, we also notice the overfitting effect for SCFL after more than sufficient number of unlabeled data are added. The learning curves usually reach their top among the first 4 iterations. Deciding the optimal stopping criteria can also be based on validation set performance but is left to future work.

To further investigate the proposed algorithm, Table 1 lists a more comprehensive comparison for all available concepts. All the results are reported in the final iteration of various algorithms. Co-training usually bears a considerable performance loss over baseline for all concepts except "Rock". On average, the average precision degrades 30%, from 21.6% to 17.7%. In contrast, SCFL is superior to the baseline algorithm in terms of average precision over all concepts except "beach". On average, the mean average precision of SCFL increases from 21.6% to 23.0%. By using additional features with unlabeled data, SCFL-M achieves even more improvement which cannot be obtained by co-training. Note that better results can be achieved if we find a better early stopping criterion. In order to show how effective SCFL is and how much potential we can gain from unlabeled data, we also show the performance of **FullLabel** in this table. As can be seen, if the additional unlabeled data was fully labeled, the mean average precision that could be achieved is 24.1%. In some sense this is the upper bound of what any labeled-unlabeled scheme can hope to achieve. This indicates that SCFL not only leverages unlabeled data with labeled data judiciously, it also achieves performance that is close to the optimal that could be achieved if the unlabeled data were to be manually annotated and used in conjunction with the labeled data. Note that sometimes adding more training data is not necessarily improving the performance, such as the concept "beach". This again shows the difficulties of video concept detection.

5. Conclusion

In this paper, we have presented a semi-supervised learning algorithm called semi-supervised cross feature learning(SCFL). Compared to co-training, the robustness and extensibility of SCFL make it more suitable for detecting the video semantic concepts. Our experiments based on the TRECVID03 concept extraction task demonstrated that SCFL not only significantly outperforms the conventional co-training algorithms, but also comes close to achieving the performance that could be obtained if the unlabeled set were to be manually annotated and used for training along with the labeled data set. The co-training algorithm on the other hand usually degrades performance compared to the baseline algorithm. The multi-view extension of SCFL can further improve upon SCFL by introducing more visual features in the unlabeled data. In the future, we can explore a

Concepts	Baseline	CoTrain	SCFL	SCFL-M	FullLabel
Airplane	0.107	0.059(-44%)	0.114(+6%)	0.112(+4%)	0.146(+36%)
Basketball	0.655	0.637(-3%)	0.670(+2%)	0.664(+1%)	0.681(+4%)
Beach	0.076	0.037(-51%)	0.071(-6%)	0.077(+2%)	0.072(-5%)
Bill-Clinton	0.111	0.097(-13%)	0.111(+0%)	0.123(+11%)	0.127(+15%)
Boat	0.087	0.014(-84%)	0.111(+27%)	0.120(+37%)	0.097(+12%)
Animal	0.195	0.132(-32%)	0.201(+3%)	0.195(+0%)	0.225(+15%)
Hockey	0.399	0.282(-29%)	0.424(+6%)	0.423(+6%)	0.439(+10%)
Mountain	0.092	0.058(-38%)	0.094(+2%)	0.102(+11%)	0.105(+13%)
Rock	0.379	0.427(+13%)	0.440(+16%)	0.443(+17%)	0.461(+23%)
Tree	0.101	0.091(-10%)	0.108(+7%)	0.110(+9%)	0.106(+5%)
Water-Body	0.177	0.115(-35%)	0.179(+1%)	0.190(+7%)	0.194(+10%)
Average	0.216	0.177(-30%)	0.230(+7%)	0.233(+10%)	0.241(+12%)

Table 1: Comparison of average precision over all eleven concepts. We compared five different algorithms of which the details are shown in Section 4.2. The results of **CoTrain**, **SCFL** and **SCFL-M** are reported in their final iteration. The percentages in the parentheses show how much improvement/degradation each algorithm can bring over the baseline.

better way to decide the stopping point for the SCFL algorithm. Finally we will extend our algorithm to other real-world applications such as video search and retrieval using a few labeled samples for querying. As future work, we can explore the possibility of plugging more advanced visual features into SCFL to improve the concept detection performance.

References

[1] TRECVID: TREC Video Retrieval Evaluation, "http://www-nlpir.nist.gov/projects/trecvid," .

[2] A. Amir, M. Berg, S. F. Chang, and et al, "IBM research TRECVID-2003 video retrieval system," in *NIST TRECVID-2003*, Nov 2003.

[3] A. G. Hauptmann and et al, "Informedia at TRECVID 2003: Analyzing and searching broadcast news video," in *Proc. of TRECVID*, 2003.

[4] M. R. Naphade, I. Kozintsev, and T. S. Huang, "On probabilistic semantic video indexing," in *Proceedings of Neural Information Processing Systems*, Denver, CO, Nov. 2000, vol. 13, pp. 967–973.

[5] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. Blei, and M. Jordan, "Matching words and pictures," *Journal of Machine Learning Research*, vol. 3, 2002.

[6] C. Lin, B. Tseng, and J. Smith, "VideoAnnEx: IBM MPEG-7 annotation tool for multimedia indexing and concept learning," in *IEEE International Conference on Multimedia and Expo*, 2003.

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

[8] K. Nigam and R. Ghani, "Analyzing the effectiveness and applicability of co-training," in *Proc. of CIKM*, 2000, pp. 86–93.

[9] M. Collins and Y. Singer, "Unsupervised models for named entity classification," in *Proc. of EMNLP*, 1999.

[10] D. Pierce and C. Cardie, "Limitations of co-training for natural language learning from large datasets," in *Proc. of EMNLP*, 2001.

[11] A. Levin, P. Viola, and Y. Freund, "Unsupervised improvement of visual detectors using cotraining," in *Proc. of the Intl. Conf. on Computer Vision*, 2003.

[12] I. Muslea, S. Minton, and C. A. Knoblock, "Active semi-supervised learning = robust multi-view learning," in *Proc. of Intl. Conf. on Machine Learning*, 2002.

[13] V.N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1995.

[14] T. Joachims, "Making large-scale support vector machine learning practical," in *Advances in Kernel Methods: Support Vector Machines*, A. Smola B. Schölkopf, C. Burges, Ed. MIT Press, Cambridge, MA, 1998.

[15] W. T. Vetterling W. H. Press, S. A. Teukolsky and B. P. Flannery., *Numerical recipes in C - 2nd ed.*, Cambridge University Press, Cambridge, NY, USA, 1994.