Large Scale Incremental Web Video Categorization

Xu Zhang[1, 2], Yi-Cheng Song[1,2], Juan Cao[1], Yong-Dong Zhang[1], Jin-Tao Li[1]

[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
[2]Graduate School of the Chinese Academy of Sciences, Beijing 100039, China
{zhangxu, songyicheng, caojuan, zhyd,jtli}@ict.ac.cn

ABSTRACT

With the advent of video sharing websites, the amount of videos on the internet grows rapidly. Web video categorization is an efficient methodology for organizing the huge amount of videos. In this paper we investigate the characteristics of web videos, and make two contributions for the large scale incremental web video categorization. First, we develop an effective semantic feature space Concept Collection for Web Video with Categorization Distinguishability (CCWV-CD), which is consisted of concepts with small semantic gap, and the concept correlations are diffused by a novel Wikipedia Propagation (WP) method. Second, we propose an incremental support vector machine with fixed number of support vectors (n-ISVM) for large scale incremental learning. To evaluate the performance of CCWV-CD, WP and n-ISVM, we conduct extensive experiments on the dataset of 80,021 most representative videos on a video sharing website. The experiment results show that the CCWV-CD and WP is more representative for web videos, and the n-ISVM algorithm greatly improves the efficiency in the situation of incremental learning.

Categories and Subject Descriptors

H.2.4 [**Systems**]: *Multimedia databases*; H.3.4 [**Systems and Software**]: *Performance evaluation (efficiency and effectiveness)*; H.E.1 [**Content Analysis and Indexing**]: *Indexing methods*

General Terms

Algorithms, Performance, Experimentation

Keywords

Large Scale, Incremental learning, Concept collection, Similarity measurement, n-ISVM, Web Video Categorization

1. INTRODUCTION

With the advent of Web 2.0, video sharing web sites become popular in recent years. As a result, the amount of videos on the internet jumps to billions and keeps growing rapidly every day. YouTube is one of the most successful video sharing web sites, with nearly 45,000,000 videos in repository and more than 100 million ones being viewed every day [1]. According to [2], YouTube has 65,000 new videos uploaded every day. Moreover, YouTube currently has about 2.86 million registered users, who are renewing the video's static information such as tags, views and links all the time.

It becomes critical to find ways to organize this large amount of data for many applications [6,12,31], such as video retrieval [3], browsing [4] and recommendation [5]. One possible methodology for organizing videos is to separate videos by category, which is widely utilized by many industrial websites such as YouTube [1] and Yahoo! Video [7]. The category information of videos on the internet is usually provided by users and further checked by human editors, which costs much time and human labor. Hence, it is important to develop a scalable algorithm for effective and efficient automatic video categorization.

We summarize three characteristics of web videos which distinguish web video categorization from traditional video categorizations as follows.

1) **Large scale**: The amount of web video categorization is no longer in thousands but in millions or even billions. Thus efficient but not only effective algorithms are in need.

2) **Growing trend**: Unlike the static video databases, web videos are uploaded every day by users. Videos with or without category information are growing. As a result, incremental algorithms are more suitable in this condition.

3) **Diversity content**: Web video is a kind of User Generated Content, so it includes information such as tags, viewing times and comments *ect.*. As a result, how to efficiently make use of the diversity content to represent web videos, and how to effectively measure similarities between web videos are two problems.

These characteristics have brought challenges for the web video categorization. One is the robust video representation to overcome the web videos' high diversity of quality, style, and genres described above. The other is the classifiers being able to handle the rapid expansion and updating of web dataset.

Previous research works on video categorization can be summarized into two types[8]. One type of research work pays much attention to multimodal features of videos [9, 10], such as audio, textual and visual features. Various types of features are extracted to represent videos. However, such features are not so effective for web video representations. The other type of research work focuses on categorization algorithms [11], including both supervised learning algorithms and semi-supervised learning algorithms. The purpose of this kind research work is to improve the categorization accuracy, regardless of the computation time and memory cost for training.

Confronted with the diversity content of web videos, it is not sufficient to represent web videos just by raw tag feature. Concept based representation method has been verified to be an effective method for video representation [19]. In this paper, we develop an effective semantic space called Concept Collection for Web Video with categorization distinguishing ability (CCWV-CD), which is consisted of concepts with small semantic gap. Furthermore, we propose the Wikipedia [15-17] propagation (WP) method to diffuse the concept correlation in this space. Experiment results

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
WSMC'09, October 23, 2009, Beijing, China.
Copyright 2009 ACM 978-1-60558-761-5/09/10...$10.00.

demonstrate the CCWV-CD and WP can represent web video more precisely and has achieved acceptable categorization results.

Different from previous algorithms used for video categorization, we propose an effective and efficient algorithm, which is incremental support vector machine with fixed number of support vectors (n-ISVM). The proposed algorithm maintains relatively high classifying performance as traditional support vector machine (SVM). Moreover, it costs much less memory and computation time for training, and can handle training samples incrementally. We conduct extensive experiments to compare our proposed algorithm with baseline methods on a benchmark dataset WEBV [18]. It can be seen that our proposed algorithm outperforms other methods, especially in the time cost and in the situation of incremental learning.

The rest of this paper is organized as follows. In section 2, the related work of video categorization is reviewed. CCWV-CD is introduced in section 3. In section 4, one scalable and efficient algorithm is developed for web video categorization. Experiments and results are presented in section 5. We make conclusions and propose some future works in section 6.

2. RELATED WORK

Web video categorization studies the video category problem in which the videos contain more information than standard videos such as films and news. In previous research works, many different features are extracted to represent videos, such as text from transcripts and extracted by Automatic Speech Recognition (ASR), visual features in various forms [24-26], audio features [27] and motion features. Recently, concept collection emerges as an effective method for video representation. One example is LSCOM [28], which has been widely utilized in TRECVID [29]. However, the text information of web videos available contains much noise. To our best knowledge there is no public available concept collection for web videos.

Once the videos are represented by the above mentioned features, various classification algorithms can be utilized. One type of algorithms works on domain specific datasets, which are made up of high quality videos such as films, news and documentaries. This genre of algorithms includes Gaussian mixture models (GMMs), hidden Markov models (HMMs) and support vector machine (SVM). Due to its high generalization ability, SVM is most widely used [13, 14] in TRECVID [29] in high level feature detection task also known as video annotation. The other type of algorithms works on web video datasets, which is made up of a hybrid of videos including professional videos, home videos and photo sequences with music. Few works [8] have been done on this kind of datasets. In the work of [8], SVM, GMM and Manifold Ranking [30] have been testified. And again SVM has been reported with the best performance. However, in the work of [8] the training set contains only thousands of videos. And to our best knowledge, there is few research works been done to deal with the incremental video categorization problem.

3. REPRESENTATIVE FEATURES

Traditional video categorization methods mainly restrict themselves to the low-level visuals features for the poor textual information, while web videos have vast amount of user-generated textual information, which can be leveraged to improve categorization performance. But there are many noises and errors in such web generated textual information. And each concept makes different contributions for distinguishing each category. So it is less reasonable to represent video in the raw tag space. In this section, we construct an ontology based on the titles and tags of web videos. Then we propose a propagation methodology using the encyclopedia on Wikipedia to enhance similarity measurement.

3.1 CCWV-CD Construction

We first introduce the Concept Collection for Web Video (CCWV) which is consisted of concepts with small semantic gap. Then Categorization Distinguishability (CD) is illustrated to gather those concepts with high categorization distinguishability. Finally, we fuse CCWV and CD as concept collection named CCWV-CD.

3.1.1 Concept Collection for Web Video (CCWV)

Y. Lu et al. [19] firstly illustrated that the concept with small semantic gap is easier to model and annotate. Motivated by this idea, we designed the following metric to measure the semantic gap of textual information from the web video dataset WEBV [18], which is consisted of 80,021 videos crawled from YouTube.

1) Select concepts with frequency more than 20 from WEBV, regarding the selected 5307 concepts as centers and separately collect related videos which are labeled with the corresponding concepts, and get a video set $V^k=<v_1,\ldots,v_n>$ for concept t_k.

2) For each video v_i, separately extract the textual vector space model (VSM) from its title and tag information, along with the 166-D color histogram (CH) of each keyframe.

3) Compute the textural and visual similarities between each pair of videos in set, and get two similarity matrixes M_{txt} and M_{vis}, and their mean values as follows:

$$Mean\_M_{txt} = \frac{2}{n\times(n-1)} \sum_{i,j=1;i\neq j}^{n} \text{cosine}(\text{VSM}(v_i),\text{VSM}(v_j)) \quad (3.1)$$

$$Mean\_M_{vis} = \frac{2}{n\times(n-1)} \sum_{i,j=1;i\neq j}^{n} \text{intersection}(\text{CH}(v_i),\text{CH}(v_j)) \quad (3.2)$$

4) Rank concepts by the gap score as follows:

$$gap\_score_{t_k} = Min(Mean\_M_{txt}, Mean\_M_{vis}) \quad (3.3)$$

While the *gap_score* is the quantity of the textual and visual consistence, the great score implicates the small semantic gap. Based on the semantic gap ranking, we select the top 2000 concepts from the total 5307 concepts as CCWV. The concepts with higher *gap_score* are more effective to represent video content.

3.1.2 Categorization Distinguishability (CD)

Similar to that different concepts have different semantic gap, the categorization distinguishability of different concepts varies either. In order to improve the accuracy of categorization, we propose a measurement named Categorization Distinguishability (CD) to extract the concepts with high distinguishability. For example, the concept NBA appears on the video title or tag implies that the video is very likely belongs to the sports category. But it is difficult to determine the category of the video if it just contains the concept video. It can be inferred that the categorization distinguishability of the NBA is much higher than that of video.

The categorization distinguishability of concepts is assumed to be strongly affected by these two factors:

DF (Document Frequency) measures the total appearance of the concept in the web video dataset WEBV [18].

CF (Category Frequency) calculates the number of categories the concept has appeared. The CF measurement only uses 30% of total videos with category information as training set.

If a concept has high DF and low CF, its CD is high. In other words, if a popular concept (with high DF) mainly appears in a few categories, its categorization distinguishability is high and vice versa. Therefore, the categorization distinguishability (CD) of the concept t_k can be express as follows:

$$CD_{t_k} = \frac{DF_{t_k}}{CF_{t_k} \cdot \alpha} \tag{3.4}$$

where α is empirical set to 10. While the CD is the quantity of the categorization distinguishability, the great score implicates the high categorization distinguishability of the concept.

3.1.3 CCWV-CD

In order to get concepts with small semantic gap and high categorization distinguishability, we fuse CCWV and CD and define the following CCWV-CD measurement of each concept.

$$\text{CCWV-CD}_{t_k} = \beta \cdot gap\_score_{t_k} + (1-\beta) \cdot CD_{t_k} \tag{3.5}$$

where β is empirical set as 0.4. We rank the 5307 concepts by CCWV-CD values and select the top 2000 concepts as the feature space $c[1...2000]$. Then, each video is represented by the classic Vector Space Model based on CCWV-CD.

3.2 Wikipedia Propagation (WP)

The average number of textual information is only 14.5 for one video in WEBV [18] base on CCWV-CD. This will result in the sparseness problem of the vector space model. In other words, the vector space model represents only part of the video information according to concepts of CCWV-CD. Besides, the relationship between different concepts should be considered as well. For example, a video with the title Boston Celtics should have some relationship with the concept NBA.

Wikipedia is employed to propagate the video textual information to the 2000 dimensions CCWV-CD concepts. The Wikipedia Propagation (WP) enhances the similarity between videos with different but related words. As a user-generated online encyclopedia, Wikipedia's universality and timeliness makes it covers a large amount of knowledge and sensible to the new concepts on the internet. Such characteristics make Wikipedia an appropriate choice for Vector Space Model Propagation. Recently, Wikipedia lab has release API [15] for the public to access the 243 million association relation among 3.8 million Wikipedia concepts. Experiments [15] have demonstrated that the concept relationship measurement extracted from Wikipedia database named *pfibf* [15] (Path Frequency-Inversed Backward link Frequency) outperforms traditional methods such as TF-IDF and co-occurrence analysis and is more coherent to human cognition. Thus we make use of *pfibf* to measure the relationship between two concepts.

For each input concept t_k, the Wikipedia API returns a list of related concepts and its *pfibf* values. The high *pfibf* value implies the closer relationship between two concepts and high position of the related concept in the return list. If a related concept is among one of the $c[1...2000]$, then the corresponding vector item $V[i] = \log_2(\frac{1+pfidf(t_k, c[i])}{1+position}+1)$. $pfidf(t_k, c[i])$ measures the relationship of concept t_k and its related concept $c[i]$, where *position* denotes the position of $c[i]$ in the return list.

4. CATEGORIZATION ALGORITHM

In the field of video categorization, support vector machines (SVMs) are the most widely used algorithm [8], because of the advantages of having no local minima and sparse representation. Thus we choose SVM as the baseline algorithm for web video categorization. But the standard batch SVM contains several limitations. First batch SVM requires all the training data available while training. But end users upload videos every day or even every minute, so video datasets on the internet are non-stationary. One solution for batch SVM is to retrain models with historical training data and new coming data, which is memory consuming and time consuming. Second due to the limitation of memories, batch SVM cannot handle well with large scale problem. To enable SVM to process huge size training data, some optimizations are made, which may degrade classifying accuracy. To handle the above mentioned large-scale and non-stationary problems, we propose the incremental support vector machine with fixed number of support vectors.

4.1 ISVM algorithm

The first incremental SVM (ISVM) algorithm for batch or individual incremental learning or unlearning is presented in [20]. It sets up a framework for ISVM algorithms and named as Cauwenberghs&Poggio (CP) algorithm. In [21] an analysis of an efficient implementation for individual learning of [20] is firstly presented. However, the problem of vector migrations is not thoroughly discussed in previous algorithms, which lead to implementation difficulties. In this section, we will try to deal with those issues based on the work in [21].

4.1.1 Preliminaries

We first introduce some preliminaries in this section. Given training data and their labels as set $D = \{(v_i, y_i), i = 1,2,\ldots,l\}$ where $v_i \in c[1\ldots2000], y_i = \{+1,-1\}$, the optimal separating function reduces to a linear combination of kernels on the training data $f(v) = \sum_j \alpha_j y_j K(v_j, v) + b$. The coefficients α_j are obtained by minimizing a convex quadratic objective function under constraints:

$$\min_{0 \le \alpha_j \le C} : W = \frac{1}{2}\sum_{i,j}\alpha_i K_{ij}\alpha_j - \sum_i \alpha_i + b\sum_i y_i\alpha_i \tag{4.1}$$

with Lagrange multiplier b, and symmetric positive definite kernel matrix $K_{ij} = y_i y_j K(v_i, v_j)$. The first-order conditions on W reduce to the Kuhn-Tucker (KT) conditions:

$$g_i = \frac{\partial W}{\partial \alpha_i} = \sum_j K_{ij}\alpha_j + y_i b - 1$$
$$= y_i f(v_i) - 1 \begin{cases} \ge 0; \alpha_i = 0 \\ = 0; 0 < \alpha_i < C \\ \le 0; \alpha_i = C \end{cases} \tag{4.2}$$

$$\frac{\partial W}{\partial b} = \sum_j y_j \alpha_j = 0 \tag{4.3}$$

which partition the training data set D and corresponding coefficients $\{\alpha_i, b\}, i = 1,\ldots,l$ into two categories: the SV set including the set S of unbounded support vectors strictly on the margin $(0 < \alpha_i < C, y_i f(v_i) = 1)$, and the set E of bounded support vectors $(\alpha_i = C)$; the NSV set of non-support vectors $(\alpha_i = 0)$, whose index in kernel matrix are respectively s, e and o.

4.1.2 Increasing Process

The main part of the incremental SVM algorithm is the procedure of adding one example to an existing optimal solution. As a result, the goal of incremental SVM is to find a weight assignment such that KT condition is satisfied for the enlarged dataset. When a new point v_c is added, its corresponding weight α_c is initialized as 0. When v_c should become a support vector, the weights of other points and b should be updated to obtain an optimal solution for the enlarged data set, i.e., keep their KT conditions satisfied. The KT conditions are expressed differentially as:

$$\Delta g_i = K_{ic}\Delta\alpha_c + \sum_{j\in S} K_{ij}\alpha_j + y_i\Delta b, \quad \forall i \in D \cup \{v_c\} \tag{4.4}$$

$$0 = y_c\Delta\alpha_c + \sum_{j\in S} y_j\alpha_j \tag{4.5}$$

Since $g_i \equiv 0$ for the margin vector set $S = \{s_1,\ldots,s_{l_S}\}$, the changes in coefficients must satisfy

$$Q\cdot\begin{bmatrix}\Delta b\\ \Delta\alpha_{s_1}\\ \vdots\\ \Delta\alpha_{s_{l_S}}\end{bmatrix} = -\begin{bmatrix}y_c\\ K_{s_1c}\\ \vdots\\ K_{s_{l_S}c}\end{bmatrix}\Delta\alpha_c \tag{4.6}$$

with symmetric but not positive-define Jacobian Q

$$Q = \begin{bmatrix}0 & y_{s_1} & \ldots & y_{s_{l_S}}\\ y_{s_1} & K_{s_1s_1} & \ldots & K_{s_1s_{l_S}}\\ \vdots & \vdots & \ddots & \vdots\\ y_{s_{l_S}} & K_{s_{l_S}s_1} & \ldots & Ks_{l_S}s_{l_S}\end{bmatrix} \tag{4.7}$$

Thus

$$\Delta b = \beta\Delta\alpha_c \tag{4.8}$$

$$\Delta\alpha_j = \beta_j\Delta\alpha_c, \quad \forall j \in D \tag{4.9}$$

with coefficient sensitivities given by

$$\begin{bmatrix}\beta\\ \beta_{s_1}\\ \vdots\\ \beta_{s_{l_S}}\end{bmatrix} = -\mathbb{R}\cdot\begin{bmatrix}y_c\\ K_{s_1c}\\ \vdots\\ K_{s_{l_S}c}\end{bmatrix} \tag{4.10}$$

where $R = Q^{-1}$, and $\beta_j = 0$ for all j outside S. Substituted in equation(4.4), the margins change according to:

$$\Delta g_i = \gamma_i\Delta\alpha_c, \quad \forall i \in D \cup \{v_c\} \tag{4.11}$$

With margin sensitivities

$$\gamma_i = K_{ic} + \sum_{j\in S} K_{ij}\beta_j + y_i\beta, \quad \forall i \notin \beta \tag{4.12}$$

and $\gamma_i \equiv 0$ for all i in S.

4.1.3 Matrix Update

To add point v_c to the working margin vector set S, $\mathbb{R}$ is expanded as:

$$\mathbb{R} \leftarrow \begin{bmatrix}\mathbb{R} & \begin{matrix}0\\ \vdots\\ 0\end{matrix}\\ \begin{matrix}0 & \ldots & 0\end{matrix} & 0\end{bmatrix} + \frac{1}{\gamma_c}\begin{bmatrix}\beta\\ \beta_{s_1}\\ \vdots\\ \beta_{s_{l_S}}\end{bmatrix}\cdot\left[\beta, \beta_{s_1},\ldots,\beta_{s_{l_S}},1\right] \tag{4.13}$$

The expansion of $\mathbb{R}$, as incremental learning itself, is reversible. To remove a margin vector k from S, $\mathbb{R}$ is constructed as:

$$\mathbb{R}_{ij} \leftarrow \mathbb{R}_{ij} - \mathbb{R}_{kk}^{-1}\mathbb{R}_{ik}\mathbb{R}_{kj}, \forall i,j \in S \cup \{0\}; i,j \neq k \tag{4.14}$$

where index 0 refers to the *b*-term.

4.1.4 Migrations between Sets and Learning

The migrations of vectors between set of S, E and O have not been thoroughly discussed in previous papers. But the main strategy for this algorithm is to identify the largest increase $\Delta\alpha_c$ such that some points migrate between sets of S, E and O. So we present a detail discussion of migrations, including incremental algorithms and decremental situation. There are four cases of possible structural changes.

(1) Migration of Support Vectors

According to equation(4.9), some α_s can reach the following limits:

Upper limits are reached for $\Delta\alpha_s \le C - \alpha_s$, which means β_s and $\Delta\alpha_c$ have the same sign, i.e. the support vector should migrate to E:

$$\Delta\alpha_c^S = \min_{s\in S}\left\{\frac{C-\alpha_s}{\beta_s}\right\} \tag{4.15}$$

Lower limits are reached for $\Delta\alpha_s \ge -\alpha_s$, which means β_s and $\Delta\alpha_c$ have opposite signs, i.e. the support vector should migrate to O:

$$\Delta\alpha_c^S = \min_{s\in S}\left\{\frac{-\alpha_s}{\beta_s}\right\} \tag{4.16}$$

(2) Migration of Bounded and Non Support Vectors

According to equation(4.11), some $\Delta g_i, i \in \{E \cup O\}$ can reach zero in the following cases:

Non support vectors, $g_o > 0$, migrations to S takes place when $\Delta g_o < 0$, *i.e.* γ_o and $\Delta\alpha_c$ have different signs:

$$\Delta\alpha_c^O = \min_{o\in O}\left\{\frac{-g_o}{\gamma_o}\right\} \tag{4.17}$$

Bounded support vectors, $g_e < 0$, migrations to S takes place when $\Delta g_e > 0$, *i.e.* γ_e and $\Delta\alpha_c$ have the same sign:

$$\Delta\alpha_c^E = \min_{e\in E}\left\{\frac{-g_e}{\gamma_e}\right\} \quad (4.18)$$

(3) Gradient

g_c becomes zero. A new vector with positive gradient is classified as support vectors belonging to S, so it will not be trained. When $g_c < 0$, $\Delta g_c = \gamma_c \Delta\alpha_c \le -g_c$ which is positive, so zero can be reached when γ_c and $\Delta\alpha_c$ have the same sign:

$$\Delta\alpha_c^g = \frac{-g_c}{\gamma_c} \quad (4.19)$$

(4) Threshold

α_c reaches C. The largest possible increment $\Delta\alpha_c^\alpha$ is:

$$\Delta\alpha_c^\alpha = C - \alpha_c \quad (4.20)$$

Finally, the maximum increment of α_c is

$$\Delta\alpha_c^{\max} = \min\left(\Delta\alpha_c^S, \Delta\alpha_c^O, \Delta\alpha_c^E, \Delta\alpha_c^g, \Delta\alpha_c^\alpha\right) \quad (4.21)$$

Once the maximum is determined, the specific vectors will migrate between sets.

4.1.5 Incremental SVM Algorithm

Let $l \to l+1$, by adding point v_c to D: $D^{l+1} = D^l \cup \{v_c\}$. Then the new solution $\{\alpha_i^{l+1}, b^{l+1}\}, i = 1,\ldots,l+1$ is expressed in terms of the present solution $\{\alpha_i^l, b^l\}, i = 1,\ldots,l$, the present Jacobian inverse $\mathbb{R}$, and the new point v_c, y_c as:

Algorithm 1 (Incremental Learning, $l \to l+1$)

1:Read point v_c, initialize $\alpha_c = 0$, compute g_c according to(4.2).
2:**while** $g_c < 0$ & $\alpha_c < c$ **do**
3: Compute β, γ according to(4.10),(4.12).
4: Compute $\Delta\alpha_c^S, \Delta\alpha_c^O, \Delta\alpha_c^E, \Delta\alpha_c^g, \Delta\alpha_c^\alpha$ according to(4.15)-(4.20).
5: Compute $\Delta\alpha_c^{\max}$ according to(4.21).
6: $\alpha_c \leftarrow \alpha_c + \Delta\alpha_c^{\max}$.
7: $\alpha_s \leftarrow \beta\Delta\alpha_c^{\max}$.
8: $g_{c,e,o} \leftarrow \gamma\Delta\alpha_c^{\max}$.
9: Let
9: **if** $k \in S$ **then**
10: Move k from S to E or O.
11: **else if** $k \in E \cup O$ **then**
12: Move k from E to S or O.
13: **else** $k == c$
14: Algorithm 1 terminates.
15: **end if**
16: Update $\mathbb{R}$ accordingly.
17:**end while**

Old vectors, from previously seen training data, may change status along the way, but the process of adding the training data v_c to the solution converges in a finite number of steps. This update process takes $O(l_S^2)$ time and $O(l_S l)$ memory.

4.2 n-ISVM algorithm

The above ISVM algorithm handles well with the increasing training samples one by one by dealing with the migration of support vectors iteratively. The number of support vectors increase dramatically with the increase of training samples. This is verified in [21] and by our experiment results in section 5.3. As a result, the cost of CPU time increases with the amount of support vectors accordingly. To reduce the running time for training and obtain good categorization results, we propose the following improved algorithm which is more efficient.

We intend to control the computation time by limiting the number of support vectors in ISVM to a fixed value n, so our algorithm is named as n-ISVM. There are two key points for n-ISVM. The first point is to determine which support vectors should remain. The second point is to decide how many support vectors should be kept.

To decide which vectors should remain, we define the following criterion to evaluate the effectiveness of each support vector.

$$c_i = y_i f(v_i) \quad (4.22)$$

The more the value of c_i approaches one, the further the vector is from the separating hyperplane, so the less effective the support vector is. After one sample is added into support vector set, all the support vectors are sorted according to its c_i value. And we remove the support vectors with large c_i from *SV* set to *NSV* set.

Algorithm 2 (LOOCV)

Given CV set *T*, get SVM model using incremental or batch SVM.
for each sample in *T*
Calculate c_i according to(4.22).
end for
Split T into *SV* set and *NSV* set.
for each sample in *SV* set
Remove one sample out of *SV* set.
while there is migrations between *SV* set and *NSV* set **do**
Perform Step 3-5 in algorithm 1.
end while
Classifying the sample using the final SVM model.
end for
Calculate the error rate.

The decrease of the number of support vectors may lead to the degradation of the performance of classifiers. Thus the value of n should be picked critically. One possible way to determine the value of n is cross validation. The classical method for cross validation is leave-one-out cross validation (LOOCV). But traditional LOOCV method for batch SVM costs much time. Here we make use of the opposite of ISVM algorithm, *i.e.* decremental algorithm to perform LOOCV as in algorithm 2.

After the size of support vector is determined through LOOCV, the n-ISVM algorithm is described in algorithm 3.

As can be seen in algorithm 3, the main procedure is essentially the same as algorithm 1. In the situation of large scale web video categorization, the number of training samples increases to tens of thousands and the value of l_S is far larger than n .But after adding the constraint of the number of support vectors, the training time cost $O(l_S^2)$ is fixed as $O(n^2)$. So in the n-ISVM algorithm, the training time will not increase with the growth of training data.

Algorithm 3 n-ISVM algorithm

Read point v_c .

Perform Step 2-4 in algorithm 1.

Sort *SV* according to c.

if *sizeof*(SV) > n

Move the least effective SV from SV set to NSV set.

else if

Algorithm 3 terminates.

end if

5. EXPERIMENT

To evaluate the representative ability of our proposed concept collection CCDW-CD, the effectiveness of the WP methodology, and the scalability of the n-ISVM algorithm, we design and carry out three experiments. The first two experiments are to evaluate the performance in the aspect of features and algorithms. The third experiment is designed to analyze the scalability of n-ISVM. All these experiments are carried on the data set of WEBV [18] as introduced in section 5.1.

Table 1. Category information of WEBV

Categories	Distribution (Percentage)	Total	Average Length (second)
Nonprofits &Activism	3.14%	2514	297
Travel &Events	3.38%	2703	220
Pets &Animals	3.77%	3014	136
Education	4.35%	3484	387
Autos &Vehicles	4.11%	3292	195
Science &Technology	4.7%	3761	304
Gaming	5.14%	4115	272
Sports	5.16%	4131	180
Howto&Style	5.74%	4590	318
Film &Animation	6.21%	4969	219
News &Politics	7.32%	5856	305
People &Blogs	9.52%	7620	247
Comedy	10.84%	8671	172
Music	11.5%	9204	223
Entertainment	15.12%	12097	205

5.1 Data Set

WEBV [18] is consisted of 80,021 most representative videos of 15 categories on YouTube in the format of FLV. The category information of WEBV is listed in Table 1, including category distributions and details of each category.

As can be seen from table 1, the distribution of WEBV is almost the same as on YouTube [22]. This is an important reason why we choose this dataset for the experiments. In this condition, the results of experiments on this dataset can demonstrate the performance of our features and algorithms on real world dataset. Moreover, labels annotated by end-users and editors of YouTube are employed as the ground-truth to evaluate the performances. As a result, no additional human labor is needed in our experiment.

5.2 Evaluation Measures

The most commonly used criteria for evaluating the performance of categorization algorithm is average precision (AP). Its definition is as follows.

$$AP = \frac{1}{N_p}\sum_{i=1}^{N} Prec_i \tag{5.1}$$

where N_p is the number of positive samples, N is the total of test samples, $Prec_i$ is the precision at cut-off rank i and defined as

$$Prec_i = \frac{\sum_{j=1}^{i} y_j \hat{y}_j}{\sum_{j=1}^{i} \hat{y}_j} \tag{5.2}$$

where y_j and $\hat{y}_j \in \{0,1\}$ is the true label and the predicted label of the ith sample. The APs of all categories are averaged as Mean Average Precision (MAP).

5.3 Performance Analysis

To evaluate the performance of CCWV-CD and WP, and n-ISVM, two experiments are carried out. In section 5.3.1 we examine the representative ability of CCWV-CD and WP. The performance of n-ISVM algorithm is testified in section 5.3.2, and compared with batch-SVM and ISVM.

5.3.1 Performance of features

In order to better understanding the performance of CCWV-CD and WP to the overall task performance, we conducted a comparison study in which we use different representation of video, including DF representation (terms with high document frequency), CCWV-CD representation without WP and CCWV-CD representation with WP. Then we train the classifier, and observe the performance. We randomly spit the WEBV into 3:6:1, with 3/10 videos as training data, 6/10 as testing data and 1/10 as cross validation set to adjust the C and γ for SVM. We train one SVM for each category. The experiment results are summarized in Figure 1.

We can see that The MAP of CCWV-CD with WP (0.552) is higher than DF (0.529) and CCWV-CD without WP (0.547). We can conclude that the new introduced CCWV-CD with small semantic gap and high categorization distinguishability improve the discriminative ability of text feature. From the comparison between CCWV-CD and CCWV-CD with WP, we can observe that in most categories, CCWV-CD with WP outperforms CCWV-CD. This proves that the incorporation of concept

relationship measurement from Wikipedia could closer the relationship of related videos and finally enhances the performance of web video categorization.

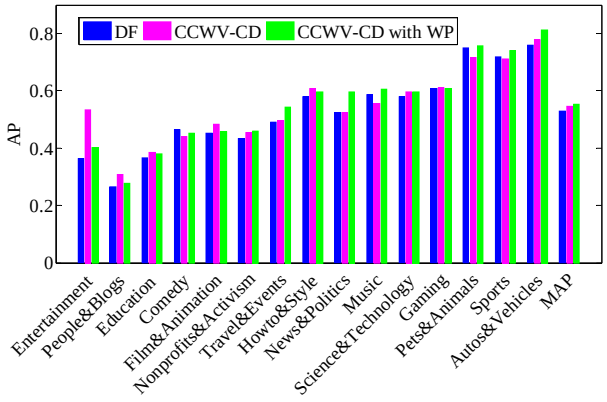

Figure 1: AP of each category using SVM

We can also observe that for news-related categories such as "News & Politics" and general knowledge related categories such as "Travel & Events", CCWV-CD with WP has gain obvious advantages with AP as 0.596 and 0.543 respectively. As explain in section 3.2, that is because as online encyclopedia, Wikipedia covers a large amount of knowledge and sensible to the new concepts on the internet. Exceptions exits for "Entertainment", "Film & Animation" and "People & Blogs", this is because there are too many proper nouns such as the film name and particular entertainment show in these categories, the incorporation of Wikipedia Propagation will introduce some noises to the samples.

5.3.2 Performance of n-ISVM

In this experiment, we compare n-ISVM with traditional SVM using LIBSVM [23] and ISVM. Since the CCWV-CD with WP has been verified to be the most effective feature in the last section, and the experiment in this section is to analyze the performance of n-ISVM, thus we choose CCWV-CD with WP as the only feature in this experiment. Three algorithms including batch SVM, ISVM and n-ISVM are compared in this experiment. The experiment results are summarized in Table 2.

Table 2. MAP on WEBV using SVM, ISVM and n-ISVM

Algorithm	SVM	ISVM	n-ISVM
MAP	0.552	0.552	0.544

It can be seen in Table 2, the MAPs of batch SVM (0.552), ISVM (0.552) and n-ISVM (0.544) are in the same level. It is reasonable that batch SVM and ISVM have equal performance, because they are solving the same optimal function in different ways. Compared with batch SVM and ISVM, n-ISVM has a degression of 0.8%. This is mainly due to the limitation of the number of support vectors. This is consistent with the conclusion in [21], that the classifying accuracy may be degreased with the decrease of the number of support vectors. But the degression in MAP results in high scalability as the results in section 5.4.

5.4 Scalability Analysis

As can be observed from the experiment result in the last section n-ISVM has some degression in performance compared with ISVM and batch SVM. However, n-ISVM can handle the training samples incrementally. Thus n-ISVM is more preferable while dealing with the web scale video categorization problem.

Here we use the WEBV data again as an example to examine the scalability of n-ISVM. In particular, we increase the number of training samples from 1000 to 10000 and record the average computation time for 15 categories on an Intel Pentium 3.20GHz and 2GB desktop. The average computation time is plotted in Figure 2, where $batch-SVM_t$ denotes the time for constructing SVM and $ISVM_t$ and $n-ISVM_t$ denote the additional time after an incremental sample is added.

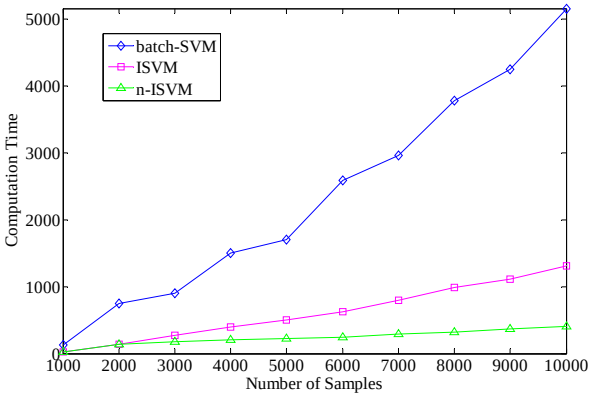

Figure 2: Training Time on WEBV data

The computation time of each method is almost linear with respect to the number of training samples. However, $batch-SVM_t$ increases much faster than $ISVM_t$. In particular, when the training sample scales to 10000, the additional time needed to construct n-ISVM is 406 seconds whereas $batch-SVM_t$ is thirteen times as much as 5162 seconds. Therefore n-ISVM algorithm outperforms both SVM and ISVM with respect to efficiency, and is more favorable in large scale incremental video categorization system.

6. CONCLUSION AND FUTURE WORK

Web video categorization contains several characteristics which distinguish it from traditional video categorizations. One characteristic is the diversity of content, format and length of web videos. The second is the large scale of the web videos. The last is that web videos arrive sequentially in day or in minute. So the robust feature is needed to overcome the web video's diversities, and the scalable algorithm is needed to satisfy the web video's rapid expansion.

In this paper, we propose a web-video-oriented concept collection CCWV-CD, and each web video is represented as a vector corresponding to CCWV-CD with Wikipedia propagation (WP). It can be seen from experiments that the new concept collection provides better representations than other features. Meanwhile, a new incremental learning algorithm n-ISVM is established by analyzing the migrations of vectors between support vector set and non-support vector set with one increasing vector. The n-ISVM algorithm can scale with tens of thousands of training samples with dimension size of 2000, and save much time for training.

The problem of how to efficiently combine the web generated text feature with the traditional visual feature is one remaining problem. For the n-ISVM algorithm, pre-decide whether the new coming vector is a possible support vector is one improvement for the future work.

7. ACKNOWLEDGMENTS

This work was supported by National Basic Research Program of China (973 Program, 2007CB311100), National High Technology and Research Development Program of China (863 Program, 2007AA01Z416), National Nature Science Foundation of China (60873165, 60802028).

8. REFERENCES

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

[2] "YouTube serves up 100 million videos a day online,". http://www.usatoday.com/tech/news/2006-07-16-youtube-views x.htm.

[3] G. H. Alexander and G. C. Michael. Successful approaches in the TREC video retrieval evaluations. In *Proceedings of the ACM International Conference on Multimedia*, 2004.

[4] O. JungHwan and A. H. Kien. Efficient and cost-effective techniques for browsing and indexing large video databases. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2000.

[5] Y. Bo, M. Tao, H. Xian-Sheng, Y. Linjun, Y. Shi-Qiang, and L. Mingjing. Online video recommendation based on multimodal fusion and relevance feedback. In *Proceedings of the ACM international Conference on Image and video Retrieval*, 2007.

[6] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo and Y. Zheng. NUS-WIDE: A Real-World Web Image Database from National University of Singapore. *ACM International Conference on Image and Video Retrieval*. Greece, 2009.

[7] http://video.search.yahoo.com/video.

[8] Y. Linjun, L. Jiemin, Y. Xiaokang, and H. Xian-Sheng. Multi-modality web video categorization. In *Proceedings of the International Workshop on Multimedia Information Retrieval*, 2007.

[9] D. Brezeale and D. J. Cook. Automatic Video Classification: A Survey of the Literature. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 38(3): 416-430, 2008.

[10] Z. Rasheed, Y. Sheikh, and M. Shah. On the use of computable features for film classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(1): 52-64, 2005.

[11] T. Ba Tu and C. Dorai. Automatic genre identification for content-based video categorization. In *Pattern Recognition*, 4:230-233, 2000.

[12] Cees G. M. Snoek , Marcel Worring. Multimodal Video Indexing: A Review of the State-of-the-art. Multimedia Tools and Applications, 25(1), 5-35, January 2005.

[13] S. F. Chang, Winston, W. Jiang, L. Kennedy, D. Xu, A. Yanagawa, and E. Zavesky. Columbia University TRECVID-2006 Video Search and High-Level Feature Extraction. In *TREC Video Retrieval Evaluation Online Proceedings*, 2006.

[14] Y.-D. Zhang, S. Tang, J.T. Li, M. Li, N. Cai, X. Zhang, K. Tao, L. Tan, S.X. Xu, Y.Y. Ran. TRECVID 2007 High-Level Feature Extraction By MCG-ICT-CAS. In *TREC Video Retrieval Evaluation Online Proceedings*, 2007.

[15] K. a. H. Nakayama, T. and Nishio, S. Wikipedia Mining - Wikipedia as a Corpus for Knowledge Extraction, In *Proceedings of Annual Wikipedia Conference*, 2008.

[16] B. Susanne. MultiTube--Where Web 2.0 and Multimedia Could Meet. *IEEE MultiMedia*, 14(1): 9-13, 2007.

[17] http://wikipedia-lab.org/en/index.php/Wikipedia_API.

[18] J. Cao, Y.D. Zhang, Y.C. Song, Z.N. Chen, X. Zhang, and J.T. Li. MCG-WEBV: A Benchmark Dataset for Web Video Analysis. Technical Report, MCG-ICT-CAS-09-001, Institute of Computing Technology, May 2009.

[19] L. Yijuan, Z. Lei, T. Qi, and M. Wei-Ying. What are the high-level concepts with small semantic gaps?. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[20] G. Cauwenberghs and T. Poggio. Incremental and Decremental Support Vector Machine Learning. In *NIPS*, 2000.

[21] L. Pavel, G. Christian, K. Stefan, ger, M. Klaus-Robert, and ller. Incremental Support Vector Learning: Analysis, Implementation and Applications. *J. Mach. Learn. Res.*, 7: 1909-1936, 2006.

[22] YouTube Report 2009. http://youtubereport2009.com/.

[23] http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[24] M. Campbell, A. Haubold, M. Liu, A. P. Natsev, J. R. Smith, J. Tešić, L. Xie, R. Yan and J. Yang. IBM Research TRECVID-2007 Video Retrieval System. In *TREC Video Retrieval Evaluation Online Proceedings*, 2007.

[25] J. Huang, S. Kumar, M. Mitra,W. Zhu, and R. Zabih. Spatial color indexing and applications. *International Journal of Computer Vision*, 35(3):245–268, 1999.

[26] D. Lowe. Distinctive image features from scale-invariant keypoints. *Int'l J. Computer Vision*, 2(60):91–110, 2004.

[27] T. Zhang and C.-C.J. Kuo. Audio content analysis for online audiovisual data segmentation and classification. *IEEETrans. Speech Audio Process*, 9(4):441-457,2001

[28] M. R. Naphade, J. R. Smith, J. Tesic, S. F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis. Large-scale concept ontology for multimedia. *IEEE MultiMedia*, 13(3):86–91, 2006.

[29] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVID. In *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, 2006.

[30] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Scholkopf. Learning with local and global consistency. In *Proceedings of Advances in Neural Information Processing System*, 2004.

[31] Loui, A., Luo, J., Chang, S., Ellis, D., Jiang, W., Kennedy, L., Lee, K., and Yanagawa. A. Kodak's consumer video benchmark data set: concept definition and annotation. In *Proceedings of the international Workshop on Multimedia Information Retrieval*, 2007.