Self-paced Learning for Weakly Supervised Evidence Discovery in Multimedia Event Search

Mengyi Liu1,2, Lu Jiang3, Shiguang Shan4, Alexander G. Hauptmann2
1Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing, 100190, China
2School of Computer Science, Carnegie Mellon University, 15213, USA
{mengyi.liu, shiguang.shan}@vipl.ict.ac.cn, {lujiang, alex}@cs.cmu.edu

ABSTRACT
Multimedia event detection has been receiving increasing attention in recent years. Besides recognizing an event, the discovery of evidences (which is referred to as “recounting”) is also crucial for user to better understand the searching result. Due to the difficulty of evidence annotation, only limited supervision of event labels are available for training a recounting model. To deal with the problem, we propose a weakly supervised evidence discovery method based on self-paced learning framework, which follows a learning process from easy “evidences” to gradually more complex ones, and simultaneously exploit more and more positive evidence samples from numerous weakly annotated video segments. Moreover, to evaluate our method quantitatively, we also propose two metrics, PctOverlap and F1-score, for measuring the performance of evidence localization specifically. The experiments are conducted on a subset of TRECVID MED dataset and demonstrate the promising results obtained by our method.

CCS Concepts
• Computing methodologies → Computer vision; Visual content-based indexing and retrieval; Machine learning; Unsupervised learning;

Keywords
Multimedia event search, Event recounting, Evidence discovery, Weakly supervised learning, Self-paced learning

1. INTRODUCTION
Nowadays multimedia contents have been produced and shared ubiquitous in our daily life, which has encouraged people to develop algorithms for multimedia search and analysis in various applications. As one of the most popular directions, multimedia event detection has been receiving increasing attention in recent years. Different from the atomic object or action recognition, which focus retrieving simple primitives [1], event detection aims to identify more complex scenario, for example, semantically meaningful human activities, taking place within a specific environment, and containing a number of necessary objects [2], which makes it more suitable for the purpose of multimedia search.

Due to such complexity mentioned above, only concern of recognizing an event is insufficient for user to understand the searching result thoroughly. A question that “why is this video classified as this event” is required to be answered, that is, our system should provide the exact temporal locations of several key-frames or key-shots from the whole video which contain observed evidences that lead to our decisions. This task is referred to as event recounting, where several efforts have been contributed to this field. For example, [3] adopted a semantic concept based event representation for learning a discriminative event model, and generated recounting by weighting the contribution of each individual concept to the final event classification decision score. [4] proposed to identify event oriented discriminative video segments and their descriptions with a linear SVM classifier and noise-filtered concept detectors, then user friendly concepts including objects, scenes, and speech were extracted as recounting results to generate descriptions. As event detection and recounting are two highly related task that could benefit with each other, some recent work aimed to address these two problems simultaneously. [5] introduced an evidence localization model where evidence locations were modeled as latent variables, and optimized the model via max-margin framework under the constraints on global video appearance, local evidence appearance, and the temporal structure of the evidence. [6] proposed a joint framework to optimize both event detection model and recounting model using improved Alternating Direction Method of Multiplier (ADMM) algorithm. [7] proposed a flexible deep CNN architecture named DevNet that detected pre-defined events and provided key spatio-temporal evidences at the same time. Figure 1 shows an illustration of event recounting results in an ideal multimedia search system.

Although these attempts have obtained promising results by indicating plausible observations, the event recounting task still remains a less addressed problem due to the challenge of evidence annotation, which leads to two limitations of the existing techniques. First, with only event labels of training videos, the evidential or non-evidential part are confused with each other and distinguished all based on category information, which omits some key evidences shared among different events, or even background samples. Second, without the ground truth of evidence loca-
Video (length: 2’39")

| Evidence | Time duration | Score | Related concepts |
|----------|---------------|-------|------------------|
| 00’14’’–00’38’’ | 0.87 | **Ring box; 2 or more people; going down on one knee** |
| 00’46’’–01’01’’ | 0.73 | **2 or more people; Holding ring; crying; hugging** |
| 01’22’’–01’47’’ | 0.59 | **2 and more people; hugging; kissing; cheering; clapping** |
| ... | ... | ... | ... |

Figure 1: An illustration of event recounting results in an ideal MED system. For each retrieved video sample, the temporal locations of key segments which contain observed evidences are provided with their predicted confidence score. The evidences are indicated by a list of semantic concepts detected in each key segments.

tions, there could not be a substantial and quantitative comparison among different methods. The performance of a system can only be evaluated by making subjective and qualitative judgement that whether the recounted evidences or semantic concepts are reasonable or not. In this paper, we focus on the mentioned issues and make efforts in the following two aspects: (1) We propose a weakly supervised evidence discovery method based on self-paced learning framework [7], which follows a learning process from easy “evidences” to gradually more complex ones, and simultaneously exploit more and more positive evidence samples from numerous weakly annotated video segments. (2) To evaluate our method quantitatively, we also propose two metrics, *Percentage of Overlap* (*PctOverlap*) and F1-score, for measuring the performance of evidence localization according to a small group of ground truth annotated by humans (The collection and generation of the ground truth are detailed in Section 4 below). The experiments are conducted on a subset of TRECVID MED dataset and demonstrate the promising results obtained by our method.

The rest of this paper is organized as follows. Section 2 presents the semantic concept based video representation, which can provide high-level semantic information that should benefit the evidence interpretation. Section 3 introduces the essential technique in this paper, i.e. self-paced learning framework, with its detailed formulation and optimization process. In Section 4, we provide comprehensive evaluations of the whole framework and comparisons with several highly related methods. Finally, we conclude the work and discuss possible future directions in Section 5.

### 2. SEMANTIC CONCEPT FEATURE

Low-level feature based video representation, for example, SIFT [8], STIP [9], and dense trajectory based features [10] [11], has been widely used in action recognition and event detection. However, those low-level features hardly have semantic meanings, thus are not suitable for interpretation purposes [5], such as the recounting task, which requires some higher-level information for event or evidence description. Recently, semantic representations based on these kind of attributes or concepts have been increasingly popular in the field of event detection and recounting [1] [4] [5] [12] [13] [14]. With the same spirit, we also learn to generate a video representation based on various semantic concepts, such as objects, scenes, and activities.

Specifically, we pre-defined our concept collection \( \mathcal{C} = \{C_1, C_2, ..., C_n\} \) (The sources of these concepts is detailed in Section 4). Table 1 provides some examples of the concepts grouped by their types. For each concept, we collect training samples, i.e. video segments, from auxiliary datasets, and employ improved dense trajectory features [11] for representation. Based on the low-level features, binary linear SVM are used for training concept detector, and finally we can generate c concept detectors totally. The next step is to process testing event videos. For the purpose of identifying the key evidences temporally, we first segment each video sample into a number of shots using well-established shot boundary detection techniques [13]. For each shot, we extract the same dense trajectory features and apply all the concept detectors on this shot to obtain c confidence scores as a representation (Note that the scores should be normalized from 0 to 1). Formally, we denote the concept representation of the \( i \)-th shot from the \( k \)-th video as \( x_{i}^{(k)} \in \mathbb{R}^c \). Suppose there are \( n_k \) shots in the \( k \)-th video, the collection of all the shots can be represented as \( X_k = \{x_{i}^{(k)}\}_{i=1}^{n_k} \), where \( k = 1, 2, ..., m \), and \( m \) is the total number of videos.

### 3. SELF-PACED LEARNING

Self-paced learning [7] is a lately proposed theory inspired by the learning process of humans or animals. The idea is to learn the model gradually from easy samples to complex ones in a iteratively self-paced fashion. This theory has been widely applied to various problems, including image classification [16], visual tracking [17], segmentation [18] [19], and multimedia event detection [20] [21].

In the context of evidence recounting problem, the easy samples are video shots with high confidence scores obtained by a binary event-oriented detector. Based on these initialized training samples, our algorithm learns a gradually “mature” model by mining and appending more and more complex evidence samples iteratively according to their losses, and also adaptively determines their weights in the next iteration. Now we start to introduce the detailed problem formulation and optimization in this section.

#### 3.1 Problem Formulation

Given \( m \) video candidates with only annotation of event labels, continue to use notations in Section 2 the \( k \)-th samples can be represented as \( X_k, Y_k = \{x_{i}^{(k)}, y_{i}^{(k)}\}_{i=1}^{n_k} \), where \( x_{i}^{(k)} \) denotes the representation of the \( i \)-th shot from the \( k \)-th video, and \( y_{i}^{(k)} \in \{-1, +1\} \) denotes its label whether
Table 1: An illustration of examples of semantic concepts grouped by their types, i.e. scene / site, object / tool, person / animal, human activity, and others.

| Scene / Site | Object / Tool | Person / Animal | Human Activity | Others |
|--------------|---------------|-----------------|----------------|--------|
| Dinning room | Refrigerator  | Female news reporter | Speaking or singing | Moonlight |
| Birthday party | Helicopter | Man wearing a suit | Shooting | Religious |
| Daytime outdoor | Appliance | Laughing baby | Cheering | Bomber bombing |
| Sunny beach | Ring box | Politician | Bicycling | Airplane landing |
| Convention center | Musical instrument | Athlete | Handshaking | Background static |
| Election campaign | uniform | Seal balancing a ball | Rock climbing | Network logo |
| Hospital | vehicle | Domestic animal | Meeting | Earthquake |
| Laboratory | Machine gun | Dolphin | Sailing ship | Overlaid text |

it can be regarded as an “evidence” or not. This formulation partially agrees with the definition of Multiple Instance Learning [22, 23], that we only know the label for each “bag” but not the instances assigned to a certain “bag”. The same point with MIL is, if \( Y_k = +1 \), which indicates that this video is categorized as a certain event (i.e. \( y_i^{(k)} = +1 \)), which means that there exists at least one evidence leading to the decision. The different point with MIL is, if \( Y_k = -1 \), in most cases there are no evidence in this video, but this cannot be guaranteed since there exists some complex and confused evidences shared among different events or even background videos, while in traditional MIL framework, \( Y_k = -1 \) leads to \( y_i^{(k)} = -1 \) for all \( y_i^{(k)} \in Y_i \).

Although we cannot employ the solution for MIL problem directly, we can exploit the same idea of heuristic optimization proposed in [23], i.e. supposing all the instances have their initialized pseudo labels and seeking for the optimal hyperplane and labels alternatively. Here in our task, we introduce all shots extracted from the background videos as negative samples, and all shots from the videos labeled as a certain event as positive samples. A linear SVM is employed to train the initialized classifier, then the current samples and model parameters are served as an initialization for Self-paced Learning in the next step.

For all the video shots \( \{X_k, Y_k\}_{k=1}^N \), where \( Y_k \) is kind of pseudo label which need to be optimized during self-paced learning process. Let \( \ell(y_i^{(k)}, f(x_i^{(k)}; w, b)) \) denote the loss function which calculates the cost between the (pseudo) label \( y_i^{(k)} \) and the predicted label \( f(x_i^{(k)}; w, b) \), where \( w \) and \( b \) represents the model parameters in decision function \( f \). In SPL, the goal is to jointly learn the model parameters \( \{w, b\} \), the pseudo label \( y \) and the latent weight variable \( v \) according to the objective function \( E \):

\[
\min_{w, b, y, v} \frac{1}{2} \|w\|^2 + \sum_{k=1}^{N} \sum_{i=1}^{n_k} v_i^{(k)} \ell(y_i^{(k)}, f(x_i^{(k)}; w, b)) + g(v, \lambda)
\]

s.t. \( y \in \{-1, +1\}^N \), \( v \in [0, 1]^N \),

where \( N = \sum_{k=1}^{m} n_k \) denotes the total number of instances from \( m \) videos, \( y = [y_1^{(1)}, \ldots, y_N^{(1)}, \ldots, y_1^{(m)}, \ldots, y_N^{(m)}] \) denotes the pseudo labels for all instances, and \( v = [v_1^{(1)}, \ldots, v_N^{(1)}, \ldots, v_1^{(m)}, \ldots, v_N^{(m)}] \) denotes their weighting parameters which reflect the sample importance in training the model, \( \ell \) is the standard hinge loss of \( x_i^{(k)} \) under classifier \( f \) (in this work, we simply employ the linear SVM version), calculated from:

\[
\ell(y_i^{(k)}, f(x_i^{(k)}; w, b)) = \max\{0, 1 - y_i^{(k)} (w^T x_i^{(k)} + b)\}.
\]

More importantly, \( g(v, \lambda) \) is the regularization term called self-paced function which specifies how the sample weights are generated. Here \( \lambda (\lambda > 0) \) is a parameter for determining the learning rate. \( g(v, \lambda) \) can be defined in various forms in terms of the learning rate [21]. A conventional one proposed in [7] is based on the \( l_1 \)-norm of \( v \) as:

\[
g(v, \lambda) = -\lambda \|v\|_1 = -\lambda \sum_{k=1}^{m} \sum_{i=1}^{n_k} v_i^{(k)}. \tag{3}
\]

This regularizer is very general and has been applied to various learning tasks with different loss functions [18, 13, 21]. Up to now, we can observe that the objective function is subjected to two parts of constraints: one is the max-margin constraints inherited from traditional SVM; another one is self-paced term taking control of the pseudo labels and sample weights respectively. This objective is difficult to optimize directly due to its non-convexity. In the next subsection, we introduce the effective Cyclic Coordinate Method (CCM) [24] to solve this problem as in [7, 25, 21, 19].

3.2 Optimization

Cyclic Coordinate Method (CCM) is a kind of iterative method for non-convex optimization, in which the model variables are divided into independent blocks (two blocks in our case): (1) classifier parameters \( \{w, b\} \); (2) pseudo labels \( y \) and sample weights \( v \). We switch between the two blocks iteratively, that one block of variables can be optimized while fixing the other block. Taking the input MIL-inspired initialization, in each iteration, the alternative optimization process can be presented as follows:

Optimizing \( \{w, b\} \) while fixing \( y \) and \( v \). In this step, we fix the pseudo labels \( y \) and weight variables \( v \) as constant, then the objective [4] is updated to only represent the sum of weighted loss across all instances as \( E(w, b) \):

\[
\min_{w, b} \frac{1}{2} \|w\|^2 + \sum_{k=1}^{m} \sum_{i=1}^{n_k} v_i^{(k)} \ell(y_i^{(k)}, f(x_i^{(k)}; w, b)). \tag{4}
\]

Generally, \( \ell_i^{(k)} v_i^{(k)} \) is the discounted hinge loss of the shot instance \( x_i^{(k)} \). To simplify the solution, in conventional SPL,
Table 2: Event names with their evidential description in TRECVID MED dataset.

| Event ID & Name | Evidential Description |
|-----------------|------------------------|
| Event 221: Attempting a bike trick | Scene or site: outdoors (skate park, parking lot or street); Objects or people: person, bike, ramps, helmet, concrete floor, audience |
| Event 222: Cleaning an appliance | Scene or site: indoors (kitchen, shop); Objects or people: appliance (fridge, stove, dishwasher), sponge, cloths |
| Event 223: Dog show | Scene or site: indoors (exhibition hall) or outdoors (fairground); Objects or people: judge(s), handler(s), dogs, name tags, leash, frisbee |
| Event 225: Marriage proposal | Scene or site: outdoors or indoors; Objects or people: small box, ring, at least two people (kissing/hugging) |
| Event 226: Renovating a home | Scene or site: indoors (basement, garage) or outdoors (roof); Objects or people: tools (hammer, wrench, drill, ladder, saw), wood, cement |
| Event 227: Rock climbing | Scene or site: outdoors (in nature) or indoors (gym); Objects or people: carabiners, rope, helmet, rock formation, climbers |
| Event 228: Town hall meeting | Scene or site: indoors (stage, arena, auditorium, church fellowship hall); Objects or people: microphone, security guards or bouncers, politicians |
| Event 229: Winning a race without a vehicle | Scene or site: outdoors (field, track, road, or stadium), indoors (gym, pool); Objects or people: athletes, finish line, boundary markers, spectators |
| Event 230: Working on a metal crafts project | Scene or site: indoors or outdoors (workshop) Objects or people: metal pieces, rivets, hammer, caliper, solder, drill |

all the $v_i^{(k)}$ are forced to be binary value, i.e. 0 or 1. Thus the objective 4 degenerates to a conventional linear SVM which only considers the selected samples whose weight equals 1. However, on the other hand, this binary setting of $v$ has limited ability for balancing the positive and negative costs, since in our task there exists only few positive evidence (event) examples while a large number of negative (background) samples. To address this problem, we employ the similar idea in Exemplar-SVM [29] which introduces two regularization parameters (i.e. $C_+$ and $C_−$) to balance the effects of these two types of costs. Differently, in our formulation, there is a small set of positive samples rather than a single “exemplar”. Accordingly, we can rewrite 4 as an ESVM-like form as follows:

$$\begin{align*}
\min_{w,b} & \frac{1}{2}||w||^2 + C_+ \sum v_i^{(k)} \ell(1, f(x_i^{(k)}; w, b)) + C_− \sum v_i^{(k)} \ell(-1, f(x_i^{(k)}; w, b)) \\
\text{s.t.} & \quad y \in \{-1, +1\}^N, v \in [0, 1]^N.
\end{align*}$$ (5)

By solving 5, we can obtain $w$ as the classification hyperplane, which is going to be fixed for the next step optimization.

Optimizing $y$ and $v$ while fixing $\{w, b\}$. With the fixed classifier parameters, we can omit the $||w||^2$ term and the objective 4 becomes $E(y, v, \lambda)$:

$$\begin{align*}
\min_{y,v} & \sum_{k=1}^{m} \sum_{i=1}^{n_k} v_i^{(k)} \ell(y_i^{(k)}, f(x_i^{(k)}; w, b)) + g(v, \lambda) \\
\text{s.t.} & \quad y \in \{-1, +1\}^N, v \in [0, 1]^N.
\end{align*}$$ (6)

Based on 6, learning $y$ is independent of $v$, and also, all the pseudo labels are independent with each other in the loss function. As each label can only take two integer values $+1$ and $−1$, the global optimal solution can be achieved by enumerating $2^N$ times.

After obtaining the optimal $y$, the final task for us is to optimize $v$. Following the solution in 7, the weight $v_i^{(k)}$ for sample $x_i^{(k)}$ can be calculated by:

$$v_i^{(k)} = \begin{cases} 1, & \ell(y_i^{(k)}, f(x_i^{(k)}; w, b)) < \lambda \\ 0, & \ell(y_i^{(k)}, f(x_i^{(k)}; w, b)) \geq \lambda. \end{cases}$$ (7)

The criterion in 7 indicates that if the loss of an instance is less than the current threshold $\lambda$, which means “easy”, it will be selected for training in the next iteration, or otherwise unselected. Here $\lambda$ controls the learning pace that how many training samples should be selected at this time. As $\lambda$ increases, the tolerance of sample loss becomes larger, and more “hard” samples will be appended to the training set to learn a stronger model. Formally, we summarize the whole optimization procedure in Algorithm 1.

Algorithm 1 : Iterative optimization of Self-paced Learning

1: $\mathbf{t} = 0$
2: Initialize pseudo labels $y$ and weights $v$
3: while $\mathbf{t} < \mathbf{max\_iter}$ do
4: $\mathbf{w}^{(t+1)}, \mathbf{b}^{(t+1)} = \arg\max_{\mathbf{w}, \mathbf{b}} E(y^{(t)}, \mathbf{w}, \mathbf{b})$
5: $y^{(t+1)}, v^{(t+1)} = \arg\max_{y, v} E(y^{(t)}, v^{(t)}; \lambda)$
6: $\lambda^{(t+1)} = \lambda^{(t)} + \Delta \lambda$
7: end while
8: return $y^{(max\_iter)}, v^{(max\_iter)}$
4. EXPERIMENTS

4.1 Dataset and Protocol

We conduct our experiments on TRECVID Multimedia Event Detection dataset \[27, 28\]. Specifically, there are two sets MED13 and MED14 collected by National Institute of Standards and Technology (NIST) for the TRECVID competition. Each dataset includes 20 complex events with 10 events in common from E221 to E230. In this paper, we only take the common part for evaluation. A detailed list of event names with their evidential description is provided in Table \[2\] and Table \[3\].

Table 3: Performance in different SPL iterations based on three concept sets. (a)(b) TRECVID SIN. (c)(d) Google Sports. (e)(f) YFCC.

According to the evaluation procedure outlined by TRECVID MED task, the dataset can be divided into 3 partitions: (1) Background, which contains 4,992 background videos not belonging to any of the target events; (2) 10Ex (or 100Ex), which contains 10 (or 100) positive video examples for each event as the training samples; (3) MEDTest, which contains about 25,000 videos for testing. Here in our recounting task, we select a small number samples from each event (about 20 for each in average), and annotate the temporal evidence locations manually. To alleviate the bias of different annotators, we average the results from 8 persons to obtain the final ground truth. Note that, this annotation is only performed on test data for evaluation purpose. For training data, only event labels are available without indications of evidences.

In order to conduct quantitative evaluation, we also propose two metrics, Percentage of Overlap (PctOverlap) and F1-score, for measuring the performance of evidence localization. For better explanation, we first define the following notations: (1) \( p \) (prediction): all temporal regions with predicted scores higher than a certain threshold (0.5 in this paper); (2) \( g \) (groundtruth): all temporal regions with annotated scores 1 (or higher than a certain threshold); (3) \( o \) (overlap): intersection regions of “prediction” and “ground truth”. Based on these notations, PctOverlap = \( o/(p + g - o) \). As precision = \( p/\text{recall} = o/g \), we can have F1-score = \( 2 \ast \text{precision} \ast \text{recall})/(\text{precision} + \text{recall}) = (2 \ast o)/(p + g) \).

4.2 Parameter Settings

For semantic concept representation, we pre-train the concept detector on three auxiliary datasets: TRECVID SIN dataset (346 concepts), Google Sports dataset \[29\] (478 concepts), and Yahoo Flickr Creative Commons (YFCC) dataset \[30\] (609 concepts), and the prediction scores of these detectors are served as a feature vector of each video shot. The SPL framework is based on these three kinds of features corresponding to different concept sets. In the learning process according to Algorithm 1 we set max_iter = 50, and in each iteration, the learning pace controller \( \Delta \lambda \) is set as \( C_+ = 0.5 \) and \( C_- = 0.01 \), which follow the default settings in \[26\], and proved to be insensitive in our experiments. Table \[5\] demonstrates the performance in different SPL iterations based on three concept sets.

Table 4: Performance comparison based on different learning methods.

(a) Metric: PctOverlap

| Method     | TRECVID SIN 10Ex | 100Ex | Google Sports 10Ex | 100Ex | YFCC 10Ex | 100Ex |
|------------|------------------|-------|-------------------|-------|-----------|-------|
| RF         | 0.3723           | 0.4446| 0.3513            | 0.4373| 0.3513    | 0.4603|
| AB         | 0.4113           | 0.4571| 0.4186            | 0.4137| 0.3837    | 0.4585|
| MIL        | 0.3531           | 0.4396| 0.3996            | 0.4544| 0.3102    | 0.4258|
| SPL        | 0.4617           | 0.5028| 0.4724            | 0.4807| 0.4594    | 0.4808|

(b) Metric: F1-score

| Method     | TRECVID SIN 10Ex | 100Ex | Google Sports 10Ex | 100Ex | YFCC 10Ex | 100Ex |
|------------|------------------|-------|-------------------|-------|-----------|-------|
| RF         | 0.4798           | 0.5651| 0.5437            | 0.5592| 0.4712    | 0.5817|
| AB         | 0.5306           | 0.5738| 0.5413            | 0.5367| 0.5119    | 0.5751|
| MIL        | 0.4829           | 0.5538| 0.4630            | 0.5519| 0.4522    | 0.5502|
| SPL        | 0.5826           | 0.6136| 0.5932            | 0.5894| 0.5808    | 0.6050|

According to Table \[3\] we can observe an approximately rising trend of the performance for both PctOverlap and F1-score, as the number of iteration increases. Specifically, for TRECVID SIN, the SPL converges really fast and achieves the peak at \#iteration = 14. This phenomenon of fast convergence also appears in the 10Ex setting for YFCC (at \#iteration = 10). Another observation is about the improvements compared to BasicMIL (the details for are presented in the next subsection). For all the three concept sets, the relative improvements for 10Ex setting are much more significant than that for 100Ex, which indicates that our method possesses strong superiority in weakly super-
4.3 Performance Comparison

We compare the propose framework to three classical methods: RandomForest (RF) [31], AdaBoost (AB) [32], and BasicMIL (MIL) as in [20]. Random forest and AdaBoost are both classical approaches which introducing sample weights implicitly in training process by random or rule-based sampling, which share the similar spirit as SPL. While in BasicMIL manner, the model is trained using all samples simultaneously with equal weights. Here we perform BasicMIL using SVM for fair comparison with SPL. All of the results are shown in Table 4 and Figure 2.

According to Table 4, BasicMIL always shows the worst results due to its straight-forward manner of data usage, i.e. no sampling and equal weights. RandomForest performs better since it considers sample weights implicitly by random sampling. AdaBoost is shown to be the best baseline method, because it performs much more similar mechanism that gradually selects “hard” samples out according to the “error”, where in SPL the criterion is “loss”.

5. CONCLUSIONS

In this paper, we propose a weakly supervised evidence discovery method based on self-paced learning framework, which follows a learning process from easy “evidences” to gradually more complex ones, and simultaneously exploit more and more positive evidence samples from numerous weakly annotated video segments. Our method is evaluated
Table 5: Performance comparison of different methods after late fusion.

(a) MED10Ex

| Event ID | Event ID | PctOverlap | F1-score |
|----------|----------|------------|----------|
|          |          | RF AB MIL SPL |          | RF AB MIL SPL |          |          |
| E221     | 0.1891   | 0.2268     | 0.1990   | 0.2760     | 0.1891   | 0.2268     | 0.1990   | 0.2760     |
| E222     | 0.2824   | 0.2768     | 0.2768   | 0.1891   | 0.2824   | 0.2768     | 0.2768   | 0.1891   |
| E223     | 0.3931   | 0.4300     | 0.4300   | 0.3931   | 0.4300   | 0.4300     | 0.4300   | 0.3931   |
| E225     | 0.1793   | 0.2771     | 0.2771   | 0.1793   | 0.2771   | 0.2771     | 0.2771   | 0.1793   |
| E226     | 0.4147   | 0.3602     | 0.3602   | 0.4147   | 0.3602   | 0.3602     | 0.3602   | 0.4147   |
| E227     | 0.3741   | 0.4159     | 0.4159   | 0.3741   | 0.4159   | 0.4159     | 0.4159   | 0.3741   |
| E228     | 0.5046   | 0.6213     | 0.6213   | 0.5046   | 0.6213   | 0.6213     | 0.6213   | 0.5046   |
| E229     | 0.2477   | 0.3959     | 0.3959   | 0.2477   | 0.3959   | 0.3959     | 0.3959   | 0.2477   |
| E230     | 0.4046   | 0.5737     | 0.5737   | 0.4046   | 0.5737   | 0.5737     | 0.5737   | 0.4046   |
| Average  | 0.2947   | 0.3948     | 0.3948   | 0.2947   | 0.3948   | 0.3948     | 0.3948   | 0.2947   |

(b) MED100Ex

| Event ID | Event ID | PctOverlap | F1-score |
|----------|----------|------------|----------|
|          |          | RF AB MIL SPL |          | RF AB MIL SPL |          |          |
| E221     | 0.2947   | 0.3968     | 0.3968   | 0.2947   | 0.3968   | 0.3968     | 0.3968   | 0.2947   |
| E222     | 0.3990   | 0.3948     | 0.3948   | 0.3990   | 0.3948   | 0.3948     | 0.3948   | 0.3990   |
| E223     | 0.5990   | 0.5261     | 0.5261   | 0.5990   | 0.5261   | 0.5261     | 0.5261   | 0.5990   |
| E225     | 0.2428   | 0.3520     | 0.3520   | 0.2428   | 0.3520   | 0.3520     | 0.3520   | 0.2428   |
| E226     | 0.6039   | 0.5490     | 0.5490   | 0.6039   | 0.5490   | 0.5490     | 0.5490   | 0.6039   |
| E227     | 0.4618   | 0.5674     | 0.5674   | 0.4618   | 0.5674   | 0.5674     | 0.5674   | 0.4618   |
| E228     | 0.6875   | 0.7876     | 0.7876   | 0.6875   | 0.7876   | 0.7876     | 0.7876   | 0.6875   |
| E229     | 0.4213   | 0.4566     | 0.4566   | 0.4213   | 0.4566   | 0.4566     | 0.4566   | 0.4213   |
| E230     | 0.5115   | 0.6311     | 0.6311   | 0.5115   | 0.6311   | 0.6311     | 0.6311   | 0.5115   |
| Average  | 0.4657   | 0.5844     | 0.5844   | 0.4657   | 0.5844   | 0.5844     | 0.5844   | 0.4657   |

on TRECVID MED dataset and shows promising results both quantitatively and qualitatively. For future work, we will attempt to investigate various forms of self-paced learning function which can be effectively adapted to our specific task for further improvement.

6. REFERENCES

[1] Yu, Q., Liu, J., Cheng, H., Divakaran, A., Sawhney, H.: Multimedia event recounting with concept based representation. In: ACM Multimedia. (2012)

[2] Li, L.J., Fei-Fei, L.: What, where and who? classifying events by scene and object recognition. In: International Conference on Computer Vision. (2007)

[3] Sun, C., Burns, B., Nevatia, R., Snoek, C., Bolles, B., Myers, G., Wang, W., Yeh, E.: Isomer: Informative segment observations for multimedia event recounting. In: ACM International Conference on Multimedia Retrieval. (2014)

[4] Sun, C., Nevatia, R.: Discover: Discovering important segments for classification of video events and recounting. In: IEEE Conference on Computer Vision and Pattern Recognition. (2014)

[5] Chang, X., Yu, Y.L., Yang, Y., Hauptmann, A.G.: Searching persuasively: Joint event detection and evidence recounting with limited supervision. In: ACM Multimedia. (2015)

[6] Gan, C., Wang, N., Yang, Y., Yeung, D.Y., Hauptmann, A.G.: Devnet: A deep event network for multimedia event detection and evidence recounting. In: IEEE Conference on Computer Vision and Pattern Recognition. (2015)

[7] Kumar, M.P., Packer, B., Koller, D.: Self-paced learning for latent variable models. In: The Conference on Neural Information Processing Systems. (2010)

[8] Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2) (2004) 91–110

[9] Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: IEEE Conference on Computer Vision and Pattern Recognition. (2008)

[10] Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: IEEE Conference on Computer Vision and Pattern Recognition. (2011)

[11] Wang, H., Schmid, C.: Action recognition with improved trajectories. In: International Conference on Computer Vision. (2013)

[12] Merler, M., Huang, B., Xie, L., Hua, G., Natsev, A.: Semantic model vectors for complex video event recognition. IEEE Transaction on Multimedia 14(1) (2012) 88–101

[13] Liu, J., Yu, Q., Javed, O., Ali, S., Tamrakar, A., Divakaran, A., Cheng, H., Sawhney, H.: Video event recognition using concept attributes. In: IEEE Winter Conference on Applications of Computer Vision. (2013)

[14] Mazloom, M., Habibian, A., Snoek, C.G.: Querying for video events by semantic signatures from few examples. In: ACM Multimedia. (2013)

[15] Boreczky, J.S., Rowe, L.A.: Comparison of video shot
boundary detection techniques. Journal of Electronic Imaging 5(2) (1996) 122–128
[16] Tang, Y., Yang, Y.B., Gao, Y.: Self-paced dictionary learning for image classification. In: ACM Multimedia. (2012)
[17] Supancic, J., Ramanan, D.: Self-paced learning for long-term tracking. In: IEEE Conference on Computer Vision and Pattern Recognition. (2013)
[18] Kumar, M.P., Turki, H., Preston, D., Koller, D.: Learning specific-class segmentation from diverse data. In: International Conference on Computer Vision. (2011)
[19] Zhang, D., Meng, D., Li, C., Jiang, L., Zhao, Q., Han, J.: A self-paced multiple-instance learning framework for co-saliency detection. In: International Conference on Computer Vision. (2015)
[20] Jiang, L., Meng, D., Yu, S.I., Lan, Z., Shan, S., Hauptmann, A.: Self-paced learning with diversity. In: The Conference on Neural Information Processing Systems. (2014)
[21] Jiang, L., Meng, D., Mitamura, T., Hauptmann, A.G.: Easy samples first: Self-paced reranking for zero-example multimedia search. In: ACM Multimedia. (2014)
[22] Dietterich, T.G., Lathrop, R.H., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artificial Intelligence 89(1) (1997) 31–71
[23] Andrews, S., Tschantaridis, I., Hofmann, T.: Support vector machines for multiple-instance learning. In: The Conference on Neural Information Processing Systems. (2002)
[24] Gorski, J., Pfeuffer, F., Klamroth, K.: Biconvex sets and optimization with biconvex functions: a survey and extensions. Mathematical Methods of Operations Research 66(3) (2007) 373–407
[25] Tang, K., Ramanathan, V., Fei-Fei, L., Koller, D.: Shifting weights: Adapting object detectors from image to video. In: The Conference on Neural Information Processing Systems. (2012)
[26] Malisiewicz, T., Gupta, A., Efros, A.A.: Ensemble of exemplar-svms for object detection and beyond. In: International Conference on Computer Vision. (2011)
[27] NIST. http://nist.gov/itl/iad/mig/med13.cfm (2013)
[28] NIST. http://nist.gov/itl/iad/mig/med14.cfm (2014)
[29] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition. (2014)
[30] Thomee, B., Shamna, D.A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., Li, L.J.: The new data and new challenges in multimedia research. arXiv (2015)
[31] Breiman, L.: Random forests. Machine Learning 45(1) (2001) 5–32
[32] Friedman, J.H.: Stochastic gradient boosting. Computational Statistics & Data Analysis 38(4)
