Multi-Lingual Broadcast News Retrieval

A.G. Hauptmann[1], M.-Y. Chen[1], M. Christel[1], D. Das[1], W.-H. Lin[1], R. Yan[1], J. Yang[1], G. Backfried[2] and X. Wu[3]

[1]School of Computer Science
Carnegie Mellon University,
Pittsburgh, PA, USA

[2]Sail Labs Technology AG
Mariannengasse 14
Vienna, Austria

[3]Dept. of Computer Science
City University of Hong Kong
Kowloon, Hong Kong

Abstract

In this notebook paper we describe the technical details of the submissions to TRECVID 2006 from CMU Informedia team. We participated in the high-level feature extraction and the search (automatic and interactive) tasks. Our emphasis is on various techniques used for the search task, where our interactive runs won the first place in the interactive track and our automatic runs are also among the top performers in the automatic track.

1 High-level feature extraction

We submitted 6 runs for TRECVID 2006 high level feature evaluation, as shown in Table 1. There were 61901 labeled shots for the 39 concepts of the LSCOM-Lite set. We split those labeled shots into a training set (45963 shots) and a fusion set (15938 shots). We use the training set to train our baseline classifiers based on various combinations of low-level features. Support vector machines (SVM) with radial basis kernel function (RBF) are used in the training of baseline classifiers. Based on our experience, the parameter setting of SVM is critical to the performance. Therefore, we perform linear search of the parameter space using cross-validation to find the optimal parameters for each concept in the training set, particularly the *gamma* parameter in the kernel function and the cost parameter. In *Run 1*, using the optimal parameter setting achieves an average of 27% improvement (0.2633 to 0.3352) over the default setting in terms of the mean average precision (MAP) metric on the 39 concepts in the cross-validation experiment.

1.1 Low-level features

Our submissions of high-level feaures are based on 4 different types of low-level features: color moment feature, Gabor texture feature, local features, and text (transcript) feature. We briefly describe each of them in the following:

- **Color moment & Gabor texture**: We appreciate Columbia University provide their color and texture features to us. To generate the color moment feature, each image (key-frame) is divided into 5x5 grids, and each grid is described by the mean, standard deviation, and third root of the

Table 1: The high-level feature extraction submissions from Informedia

Run	Name	Method	Features
1	*"The Phantom Menace"*	SVM, multi-modality (early fusion)	color, texture
2	*"Attack of the Clones"*	SVM, multi-modality (late fusion)	color, texture, local feature, monolingual text
3	*"Revenge of the Sith"*	MDRF without X^2 selection	color, texture, local ceature, monolingual text
4	*"A New Hope"*	MRRD with X^2 selection	color, texture, local feature, monolingual text
5	*"The Empire Strikes Back"*	SVM, multi-modality (late fusion)	color, texture, local feature, multilingual text
6	*"Return of the Jedi"*	Borda voting	color, texture, local feature, multilingual & monolingual text

Table 2: The comparison of logistic regression and SVM in multi-modality fusion

Multi-modality Runs	Average precision
logistic regression (color-texture + local +monolingual text)	0.146
SVM (color-texture + local +monolingual text)	0.121
logistic regression (color-texture + local +multilingual text)	0.153
SVM (color-texture + local + multilingual text)	0.126

skewness of each color channel in the LUV color space. This results in a 225-dimension (5x5x3x3) color moment feature. Texture feature comes from the Gabor filter, which denotes an image by mean and standard deviation from the combination of four scales and six orientations.

- **Local features**: The local feature of each image is computed from the local interest points (keypoints) detected from the image. We use the keypoints [10] provided by City University of Hong Kong, which are detected using the DoG detector and depicted by SIFT descriptors [6]. We cluster the keypoints into 1000 clusters in their descriptor space, and represent each image by the distribution (counts) of the keypoints belonging to each cluster based on a 3x3 grid. This results in a vector-quantized local feature of 9,000 dimensions for each image. We have also experimented with other settings with different cluster numbers and grids, and the one used (1000 clusters with 3x3 grids) has achieved the best performance.

- **Text features**: Text features have been shown to successfully complement visual features in constructing effective multi-modal visual classifiers. Extracting text features on a multilingual corpus, such as TRECVID'05, however, faces an additional problem: how should we effectively combine information from multiple languages? One straightforward solution is to translate multilingual text (e.g., ASR transcripts) into a common target language (e.g., English), and we can proceed classifier learning and evaluation protocols as if there were no multiple languages. The advantage of this approach is that training examples in English are abundant. The disadvantage, however, is that automatic translation systems inevitably introduce errors in addition to errors from automatic speech recognition systems. To leverage abundant training examples and discriminative power from native languages, we propose a multilingual text features for learning text-based visual classifiers. Monolingual text features are a bag-of-words representation of words spoken in a shot of dimensions of V_E, where V_E is the vocabulary size of English. Multilingual text features, on the other hand, contain both native languages and translations (e.g., Chinese and English translation), and is of the dimension $V_E + V_C + V_A$, where V_C and V_A are the vocabulary sizes of Chinese and Arabic, respectively. We build text classifier on this multi-lingual feature using SVM with a linear kernel. We evaluated the proposed multilingual text features on the development set of TRECVID'06. Experimental results showed that multilingual text features were remarkably more effective than monolingual text features (i.e., English only). Multilingual run improved the mean average precision (MAP) of the 39 concepts from 0.134 to 0.175 (30% improvement) on a held-out test set. Contrasting our submission Run 2 and 5 also shows multilingual text features consistently perform better than monolingual text features. In addition to the ASR transcripts and translations by provided by NIST, text features were also obtained using the SAIL Labs [www.sail-technology.com] speech recognition engine for English and Arabic speech recognition. The Arabic transcripts were further translated into English using Google translation [www.google.com/translate_t] through automated scripts.

1.2 Using Multi-modal Features

Multiple types of low-level features need to be combined in an effective way to provide better performance than any single type of features. We adopt a mixture of the early fusion and late fusion strategy. To color

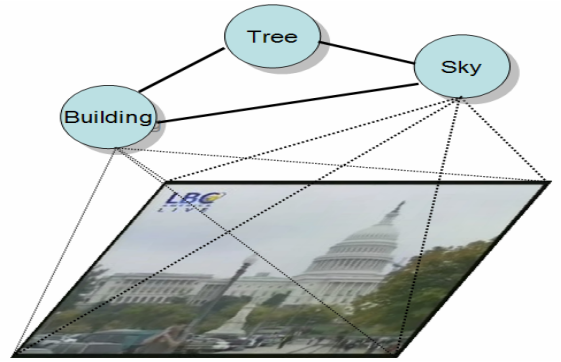

Figure 1: A graph demonstrates the framework of MDRF. There are three semantic concepts in this video shot: building, tree and sky. The top layer shows the concepts relations with each other and constitutes an undirected graph. The edges between each concept can be viewed as interaction potentials in the MDRF formula. The dotted lines from concepts to the video shot. illustrate the classifications of each concept which act as association potentials in the MDRF model. In the MDRF model, concepts are denoted as variable y and a video shot is denoted as observation X.

and texture features are stacked into a large feature vector of 273 dimensions (i.e., early fusion) due to their low dimensionality and close relationships. In contrast, we use late fusion strategy to combine this color-texture feature with the local feature and the textual feature. Specifically, we train SVM classifier for each concept based on each type of feature, and apply the trained classifiers to predict the label of each shot in the testing set. Therefore, for any shot, there will be predictions based on color-texture feature, local feature, and text feature, respectively. We train meta-level classifiers using logistic regression or SVM, which take the component prediction scores as input and output an overall prediction. Table 2 shows the comparison between the two meta-level classifiers with different low-level features. Clearly, logistic regression outperforms SVM as the meta-level classifier in this corpus. We thus choose logistic regression to fuse the predictions based on multi-modal features.

1.3 Exploiting Multiple-Concept Relationships

From our previous experience, the semantic concepts to be detected are not independent to each other. Thus, exploiting relationships between multiple semantic concepts in video could be an effective approach to enhancing the concept detection performance. In TRECVID 2006, we tried to use a multi-concept fusion technology called Multiple Discriminative Random Field (MDRF).

Figure 1 illustrates the framework of MDRF on top of a single video shot, which consists of several different semantic concepts, such as *"building"*, *"tree"*, and *"sky"*. Therefore, we can construct an undirected graphical model to represent the relationships between concepts and the video shot, and also the relationships between various concepts. Figure 2 illustrates such a graphical model. Mathematically, MDRF is stated as:

$$p(Y \mid X) = \frac{1}{Z} \exp\left(\sum_{i \in S} A_i(y_i, W, X) + \sum_{i \in S} \sum_{j \in N_i} I_{ij}(y_i, y_j, V, X) \right) \qquad (1)$$

where $Y = (y_1, y_2, \ldots, y_n)$ is the vector of multiple concept labels, with y_i denoting the label of ith concept. In this work, each semantic concept is either present or absent in the shot, i.e. $y_1 = \{-1, 1\}$. X (in $R^{(c)}$) is the observation or feature vector extracted from the video shot. $A_i(y_i, W, X)$ is called the *association potential* function. In MDRF, association potential provides links between concept labels and observation, as a normal classifier does. $I_{ij}(y_i, y_j, V, X)$ is the *interaction potential* function. The interaction potential tries to model the interactions between various concepts with observation. For example, if there are some shots in training set that have both the "*sky*" and "*tree"* concept, the bluish and greenish color feature (which are typical for the two concepts) will be emphasized in learning process via the interaction potential. Therefore,

when a new shot comes out with big blue which is easy to be recognized with unclear green area, the tree detector will get the benefit from the interaction potential to detect the tree concept. $\theta = \{W, V\}$ are the parameters in the mode. W is the parameter of the association potential, and V is the parameter of the interaction potential. In Eq.(1), the summation of association potentials corresponds to the set of individual classifiers for each concept, and the summation of interaction potentials models the relationship between each concept pair.

Now we can take a deeper look at Figure 2. In Figure 2, we can understand MDRF as a fully connected undirected graphical model. There are 3 concepts as Y_1, Y_2 *and* Y_3 which are linked to each other as well as to the observation X. The linkages between concepts encode the interaction potentials in MDRF and the linkages to observation encode the association potentials.

In TRECVID 2006, we predict fusion set and test data (the new trecvid 2006 data) by the models we built from our training set. Therefore, for shots in fusion set and test data, predictions are our observations for this shot. To be more clearly, we have 39 different concepts and every concept has 4 different modalities. The observation is a 156 dimension vector (39x6). We adopt logistic function as association potential:

$$A_i(y_i, W, X) = \log(p(y_i \mid X)) = \log\left(\sigma\left(y_i w_i^T h_i(X)\right)\right) \qquad (2)$$

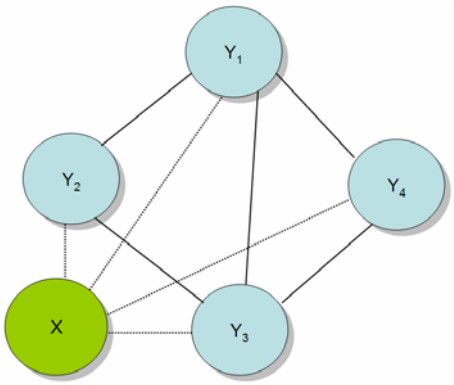

Figure 2: MDRF is a fully connected undirected graphical model. Y nodes denote the semantic concepts. X is the observation extracted from the video. All concepts are dependent on the observation.

$$I(y_i, y_j, V, X) = y_i y_j V_{ij}^T u_{ij}(X) \qquad (3)$$

From Eq.2, we know the association potential works like a logistic regression classifier which outputs the probability of label given the observation. Eq.3 shows the interaction potential function. $u_{ij}(X)$ can be any specific function to deal with the observation. V is the parameter of interaction potential, which tries to emphasize the agreement between two concepts and also to search the observation that supports the agreement.

Table 3 shows the performance of MDRF in TRECVID 2006 submission in comparison with a SVM approach that does not consider inter-concept relationships. We use feature selection method based on chi-square statistics to filter out some concept pairs which are not related in order to remove noises from the model. We discovered that even when the threshold of chi-square statistics is as small as 0.05, only very few concepts in 39 concept corpus connected to each other. It means that not many concepts are related to each other in TRECVID 2006 set; therefore, we didn't get a significant improvement by considering the multi-concept relationships. We also found that chi-square feature selection is critical since without it the performance was much lower.

Table 4: Automatic and interactive search submissions from Informedia

Run	Description
See	Full Informedia interface, expert user, query-by-text, image, concept, and auto-topic functionality
Hear	Image storyboards working only from shots-by-auto-topic (no query functionality), 2 expert users
ESP	Extreme video retrieval (XVR) using manual browsing with resizable pages (MBRP), relevance feedback, no query functionality
Smell	Extreme video retrieval (XVR) using rapid serial visual presentation (RSVP) with system controlled presentation intervals, relevance feedback, no query functionality
Touch	Automatic retrieval based on only transcript
Taste	Automatic retrieval based on transcript and all other modalities

Table 3: High-level feature extraction performance (AP) of MDRF

2 Automatic Search

Both the automatic and interactive search submissions are summarized in Table 4. Similar to last year, all of our automatic retrieval submissions are built on a relevance-based probabilistic retrieval model, which aims at combining diverse knowledge sources from different retrieval components and semantic concept outputs. This model translates the retrieval task into a supervised learning problem with the parameters learned discriminatively. Rather than treating retrieval as a classification problem, we used an algorithm called "ranking logistic regression"[1] which considers the order information between training data instead of their binary labels, so that the optimization objective is closely associated with the retrieval performance criteria (i.e., mean average precision). Specifically, this algorithm finds a set of weights used to combine relevance scores generated from various knowledge sources into an overall relevance score. In addition to the ASR transcripts and translations by provided by NIST, both automatic search and interactive search also used text features extracted using the SAIL Labs [www.saillabs.com] speech recognition engine for English and Arabic speech recognition. The Arabic transcripts were further translated into English using Google translation [www.google.com/translate_t] through automated scripts.

2.1 Query Analysis

A wealth of prior research shows that simply adopting a query-independent knowledge combination strategy is not flexible enough to handle the variations in users' information needs. It is desired to develop more advanced methods to incorporate query information into the probabilistic retrieval model. To achieve this goal, we proceed by making the following assumptions on the query space: 1) The entire query space can be described by a finite number of mixtures (query types), where the queries from each mixture have the similar characteristics and share the same combination function; 2) Query descriptions can be used to indicate which mixture (type) it belongs to [2].

As a simple approach, we define query types based on human knowledge. Formally, a probabilistic retrieval model can be represented as:

$$P(y_+ \mid D,Q) = \sum_{k=1}^{K} P(z_k \mid Q) \cdot \sigma\left(\sum_{i=0}^{N} \lambda_{ki} P\left(S_i \mid D,Q\right) \right) \quad (4)$$

where z_k is the variables indicating the defined query types. There is one and only one z_k set to 1 while the other set to 0. Similar to the setting of last year, we automatically assigned each query to each of the five defined query types [3]:

- **Named person:** queries for finding a named person, possibly with certain actions
- **Named object:** queries for a specific object with a unique name or an object with consistent visual appearance.
- **General object:** queries for a general category of objects instead of a specific one among them
- **Sports:** queries related to sport events
- **Scene:** queries depicting a scene with multiple types of objects in certain spatial relationships

The method for automatically assigning queries into the query types, namely query type classification, can be found in our previous work [3]. The parameters of the model for each query type can be estimated based on Eq.4 from the queries of that specific type and their (labeled) results. To process a new query, we first classify it into a specific query type and then use the corresponding retrieval model to find its relevant shots. This query type based retrieval model has been demonstrated to be successful in recent work [4].

3 Interactive Search

Our goals for the 2006 TRECVID interactive search task include:

- Charting the evolution of topics and corpora through the years and gauging novice versus expert performance by having the same expert user again perform the 2006 TRECVID search
- Leveraging from past successes in exploring new approaches, especially improvements in how to utilize usage context more appropriately for streamlining what the user sees and the order in which shots are presented.
- Following the recommendations from our CIVR 2006 publication by Christel and Conescu [8] from TRECVID 2005 studies on how to better utilize the implicitly overlooked shot sets (those shots skipped over when marking relevant shots) for the automatic expansion of the user's marked shot contribution up to 1000 shots.
- Examining the effects of different interfaces presenting the ranked shot output of automatic retrieval systems, particularly on the reason of the good performance of extreme video retrieval (XVR) interface (is it because XVR is an inherently better interface than the dense storyboards used in the traditional Informedia interface, or is it due to the availability of good ranked shot output from automatic retrieval?)
- Exploring the benefits of incorporating relevance feedback and active learning to reorder presentations during the course of the 15-minute interactions.

The first point is covered by having one of the Informedia researchers remain completely isolated from the TRECVID test set and topics until he performs the interactive search task, and then submit that run, as was done for 2002, 2003, 2004, and 2005. In this year, all the runs, including those based on the traditional Informedia interface, had access to the ranked shot output of the fully automatic search, which was only accessible to the XVR runs in TRECVID 2005 submission.

To facilitate easier comparison across TRECVID participants, we adopt the query access descriptions provided by Snoek and colleagues in their CIVR 2006 paper [9]: "query-by-keyword", "query-by-example", and "query-by-concept." To better distinguish the primary modality of the query, we name these access methods "query-by-text", "query-by-image", and "query-by-concept", respectively, with one addition: the available of the ranked shot output of the fully automated search, which we label "query-by-best-of-topic" since this is a topic-specific shot list.

The **See** treatment allows users to retrieve and view shots based on "query-by-best-of-topic" in addition to "query-by-text", "query-by-image", and "query-by-concept." The **Hear** treatment has the same interface as

See, except that it can only access shots by "query-by-best-of-topic", with all other query functionality (query-by-text, by-image, by-concept) inaccessible. In comparison, **See** treatment has all these query capabilities. Like **Hear**, the other 2 interactive runs, **ESP** and **Smell**, have access only to the output of the automatic search run, except that they use extreme retrieval (XVR) interfaces rather than the traditional Informedia dense storyboard displays. Figure 3 shows the interactions breakdown for the different treatment groups following an analysis of the transaction logs for the accumulation of 15-minute topic sessions. Figure 3 is interesting in that the dominance of the query-by-text strategy for the **See** interface is much less than in prior years, with the rich query functionality allowing for a diversity in interactions.

Regarding the "query-by-concept" functionality, we worked only with the LSCOM-Lite concepts, and hence all of our runs were type "A." If we had a richer palette of LSCOM concepts, the query-by-concept functionality would perhaps have received a greater percentage of attention in the **See** treatment, and perhaps also led to better ranked shot sets for the query-by-best-of-topic group, since the automatic run utilizes semantic concepts as well. This is a point for future work. The goal to "leverage from past successes in exploring new approaches" motivated our continuing investigation into the use of our fully automated search runs which performed well in 2004 and led to XVR's use in 2005. So, rather than change too many variables in our interactive search runs, we chose to remain with LSCOM-Lite and type "A" for 2006 with the investigations looking at both improved interface efficiency and the effects of alternative interfaces (i.e., MBRP, RSVP, and, for the first time, traditional Informedia storyboards) utilizing output of the automated search runs.

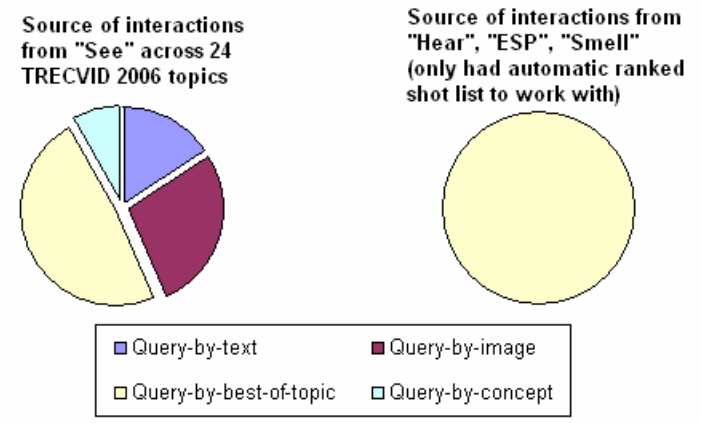

Figure 3: Breakdown of interactions (only "See" had access to rich query functionality).

The improved efficiency in the traditional Informedia interface made for TRECVID 2006 and used by the **See** and **Hear** groups resulted in impressive numbers of shots being reviewed within the 15-minute time limit. Figure 4 shows a conservative count of the number of Informedia shots either marked as correct, or passed over when marking another shot as correct, during the 15-minute topic run. In our system, we represent all of the sub-shots (NRKF) and master shots (RKF) as "Informedia shots" and at the end of the user marking, collapse down the Informedia shots to the TRECVID common shot reference. We have 146,328 Informedia shots in the test set. The review count shown in Figure 4 is conservative in that if the user finishes off with a review of a storyboard with, say, 78 irrelevant shots at the end of the storyboard that get no user action, those 78 are not counted because the user neither marked any as correct nor passed over them and marked a following one as correct. Of course, with better transaction logging we could get more accurate counts, but even these lower bound numbers are impressive in that the interface supports review of thousands of shots within the time limit. Clearly, some topics take more time to review than others, e.g., "*emergency vehicles in motion*" and "*a kiss on the cheek*" were time-consuming.

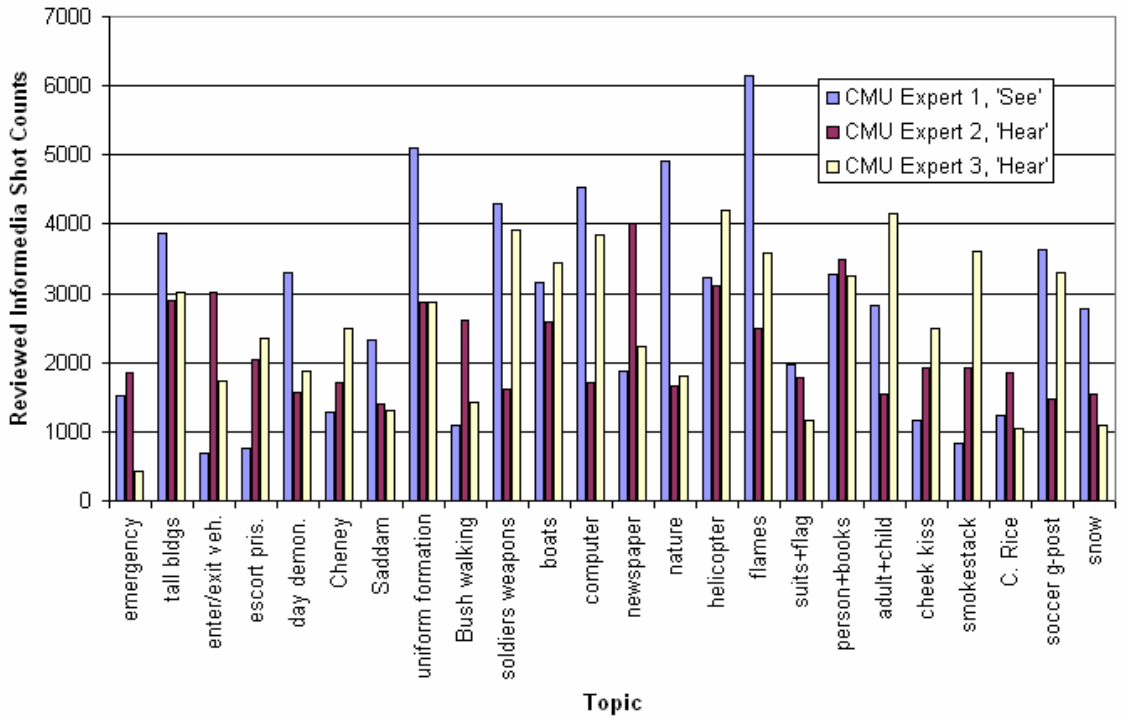

Figure 4: Number of shots reviewed per topic for 3 users of Informedia storyboard systems.

In prior TRECVIDs, the fully automated search run succeeded with good recall for many of the topics with no user in the loop, but the relevant shots were distributed throughout the top 3000 to 5000 slots in the ranked shot list, causing the average precision (AP) for the automated search run to lag well behind the AP scores for the best interactive runs. By relying on an intelligent human user possessing excellent visual perception skills to compensate for comparatively low precision in automatically classifying the visual contents of video, a human user could filter the automated set and produce a set that retains most or all of the relevant shots from the automated set, but with much greater precision. We compared three different interfaces to see which one maximizes the human efficiency in labeling relevant shots from the ranked list. The **See** and **Hear** treatments used the Informedia storyboard interface, the **ESP** and **Smell** used the extreme video retrieval (XVR) with rapid serial visual presentation (RSVP) strategy and manual browsing with resizable pages (MBRP) strategy, respectively. Given that **See**, **Hear**, and **ESP** scored as 3 of the top 5 search runs as rated by mean average precision, we conclude that the input from the automated search runs is quite useful as a starting point for interactive search. The benefits of additional interactive query capability are shown by the improved performance of **See** over the other treatments.

We acknowledge the need to test different interfaces with non-expert users as well, users who do not have familiarity with either TRECVID in general or Informedia interfaces in particular. In fact, we have run experiments with experienced intelligence analysts meeting these criteria (no prior experience with TRECVID, Informedia). Their runs, though, occurred after the deadline for TRECVID submissions and hence will not be reported here, although if the analysis can be completed in time, such runs might be discussed orally at the workshop to contrast with these runs reported here by experts in the CMU interfaces.

The average precision for the 6 CMU search runs in TRECVID 2006 are reported in Figure 5. The **See** treatment was ranked first among all TRECVID search runs, and the other 3 interactive runs (**Hear, ESP, Smell**) were ranked among the top performers. In comparison with the results of TRECVID 2005, we conclude that the quality of the input ranking list, which was from the automatic search run, is critical to the performance of interactive searches. The advantage of **See** over **Hear** shows that the various query functionalities in Informedia client provides significant additional performance improvements.

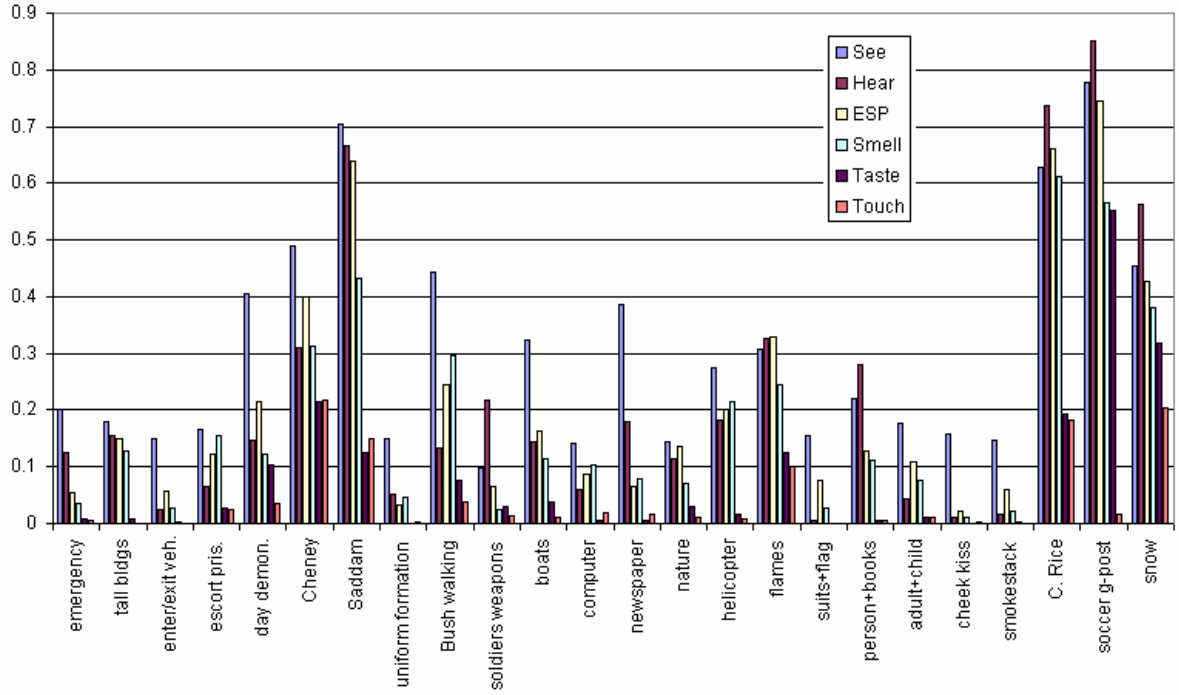

Figure 5: Average precision across TRECVID 2006 topics for CMU search runs.

3.1 Relevance Feedback

In interactive search, we can take advantage of user feedback in order to improve the ranking of relevant shots for a given user query. This was done incrementally during the search process: while the user goes down a ranked list and assigns labels to the shots in it, the system learns from the user labels and improves the remaining part of the ranked list. We implemented two relevance feedback approaches. The first updated the weights for combining retrieval scores from various knowledge sources, and the second re-ranked the shots based on their temporal proximity to marked relevant shots.

3.1.1 Re-weighting of Knowledge Sources

The retrieval model contains a set of weights used for combining the retrieval scores computed based on different knowledge sources (modalities), such as text, color, texture, motion, and semantic concepts, and the weights are specific to each query type. To process an incoming query, we first map the query into one of the query types, and then use the corresponding set of weights to compute the relevance score of the candidate shots. However, there is some query information that could not be captured this query-type representation. For example, the query "finding the maps of Baghdad" has strong hints to suggest incorporating the output from the semantic concept "maps". More examples are shown in Table 5. Given the limited number of query types, we cannot easily take this information into account. However, once we have some relevance feedback in the form of labeled training data, we can further refine the combination weights, i.e., when we find there is direct match between query descriptions and the semantic concepts, the corresponding concepts will be associated with a positive weight. In our current implementation, the concept weights are set to be equal to the weight of text retrieval.

As one of the useful techniques to improve the retrieval performance, the relevance feedback algorithm proceeds by requesting users to annotate a small number of selected video documents from the initial retrieval results and then feeding them back to update the retrieval models. Formally, we denote the relevance information as $y_1,...,y_F$ associated with the feedback documents $D_1,...,D_F$. It can be viewed as a learning component in a retrieval system, where the system learns from a small amount of relevant examples to adjust the ranking function accordingly. In this proposal we mainly consider using the additional annotated data to adjust the combination parameters λ in the probabilistic retrieval models.

Table 5: Examples of TRECVID queries and their related semantic concepts

Queries	Related Semantic Concepts
Find the *maps* of Baghdad	maps
Find one/more *cars* on the *road*	cars, roads
Find a *meeting* with a large table	meeting
Find one/more *ships* and *boats*	ship_and_boat

Given the relevance judgments, we propose the following model-based relevance feedback approach by computing the maximum a posteriori estimation for the updated combination parameters,

$$\begin{aligned}\lambda^* &= \arg\max_{\bar{\lambda}} P(\bar{\lambda} \mid y, D, Q, \lambda) \\ &= \arg\max_{\bar{\lambda}} P(\bar{\lambda} \mid \lambda) \prod_t P(y_t \mid D_t, Q, \bar{\lambda}) \\ &= \arg\max_{\bar{\lambda}} \left[\log P(\bar{\lambda} \mid \lambda) + \sum_t \log P(y_t \mid D_t, Q, \bar{\lambda}) \right]\end{aligned}$$

where λ are the initial parameters for combination and λ* are the updated parameters after relevance feedback. The prior probability can be defined in many ways and we particularly define it as a Gaussian distribution with mean λ and a pre-defined variance. This formulation can also be interpreted from the maximum likelihood estimation point of view, which is actually making the compromise between two factors: one is minimizing the distance between the updated model parameters and the initial model parameters, and the other is maximizing the likelihood for the feedback data.

We have also utilized an additional selection strategy that is computationally less complex and shows great promise for extreme retrieval. The crucial insights come from an analysis of temporal sequences in video concepts. After noticing that the semantic concept in a keyframe of a shot is the single best predictor for the concept in the next shot, we tested whether this would also hold true for query results. In other words, if we find a relevant shot, we predict that the same 'query concept' is likely to be relevant in the adjacent shot. This gives a new framework for re-ranking. When a shot in the ranked list of queries is marked as relevant by the user, we simply insert the neighbors of this shot at the top of the shots to be presented to the user on the next display page. These approaches were used in the interactive search runs ESP and Smell.

3.1.2 Temporal Re-ranking

We have also developed an alternative relevance feedback strategy that is computationally less complex and shows great promise for extreme retrieval. The crucial insights come from an analysis of temporal consistency of video shots, which shows that relevant shots of a query topic are likely to appear consecutively or close to each other. In other words, if we find a relevant shot, we predict that the same 'query topic' is likely to be relevant in the adjacent shots. This leads to a new approach for re-ranking the shots in a ranked list. Specifically, when a shot in the ranked list of queries is marked as relevant by the user, we simply insert the neighbors of this shot at the top of the shots to be presented to the user on the next display page. Clearly, the main parameter for this re-ranking based on temporal proximity is the number of adjacent neighbors to include, where we experimented with windows of 1, 2, 3, 5, and 10 shots on both sides of a shots marked as relevant by the user. We used window of 10 shots for the TRECVID 2006 submission. Extensive experiments on previous TRECVIDs data have demonstrated the effectiveness of this approach [7].

3.2 Extreme Video Retrieval

Based on the success of the extreme video retrieval (XVR) interface [5] in TRECVID 2005, we included two XVR-based interactive runs in TRECVID 2006. They are based on the strategy called rapid serial visual presentation (RSVP) and manual browsing with resizable pages (MBRP), respectively.

3.2.1 Rapid Serial Visual Presentation (RSVP)

This mode used the keyhole presentation format, with the system displaying only one image in a page and turning pages automatically at a certain speed. To help them focus, human subjects made the desktop background completely dark and removed all icons and toolbars. Initial starting speed for system image display was usually set around 400-500 ms, depending on how the subject was feeling. This time can be reduced by the subject later on in a run, down to about 200 msec per image at peak speeds. Subjects reported that they didn't really slow down at all, but instead increased speed over time. Due to the variable reaction time latency, two shots were marked as relevant for each key press (the one currently visible and the one before). There is a verification phase [4] near the end of the 15-minute limit, which allows the subject to review his labels and make corrections. This year we also added a feature called “slideshow mode”, which displays 5 key-frames (instead of one key-frame) evenly distributed along the temporal axis of a given shot. This slideshow mode allows subjects to see the dynamic information of the shot, which is critical for queries related to motion information, e.g., “emergency vehicles in motion”. The slideshow mode is activated and inactivated by pressing the “Ctrl” key.

The automatic RSVP mode was reported to be very stressful at the beginning and probably didn't give a great advantage at the beginning, though it appeared very useful at the end. It was really beneficial at the end because subjects need to look at a large number of images, each with a very low probability of having a match. It was not very helpful at the beginning because the relevant shots are more dense and subjects would often either have to stop and backtrack when they noticed a match, or when they were weren't sure they tagged an image correctly. When a query requires identifying multiple features or objects, i.e. "a person WITH at least 10 books" or "a chimney WITH smoke coming out of it", subjects found that it was usually more effective to label it (as relevant) and look at its details more carefully in the verification phase. Subjects didn't use the “slideshow mode” feature very often, except when the motion information was absolutely necessary. After the submission, we also found that marking two consecutive shots as relevant in RSVP mode, however, negatively interact with temporal re-ranking and active learning mechanism. This is because nearly half of the labeled shots are not actually relevant, which mislead the learning and re-ranking algorithm.

3.2.2 Manual Browsing with Variable, Resizable Pages (MBRP)

MBRP is an alternative strategy for interactive search, which gives the user more control of the image display. The subject decides the number of key-frames to be shown in each page, which can be changed dynamically in the course of search. Instead of letting the system turn the page, subjects press “F” and “D” key (on left hand) to page forward and backward. Since the time a user spends browsing each page depends on the page size, the visual complexity of the answer, and the number of correct shots to label on the page, this time can vary dramatically with different pages. The user may occasionally need to turn back to previous pages to correct erroneous labels. Thus MBRP gains one advantage by letting users turn pages through key press. The downside is that a conservative user might perform sub-optimally by taking too much time per page to double and triple check every selected answer.

Acknowledgements

This work was sponsored in part by the National Science Foundation trough grant NSF IIS 0535056 and by the Disruptive Technology Office (DTO/ARDA) under contracts number H98230-04-C-0406 and NBCHC040037.

References

[1] R. Yan and A. G. Hauptmann. Efficient margin-based rank learning algorithms for information retrieval. In *International Conference on Image and Video Retrieval(CIVR)*, 2006.

[2] R. Yan and A. G. Hauptmann. Probabilistic latent query analysis for combining multiple retrieval sources. In *Proceedings of the 29th international ACM SIGIR conference*, Seattle, WA, 2006.

[3] R. Yan, J. Yang, and A. G. Hauptmann. Learning query-class dependent weights in automatic video retrieval. In *Proceedings of the 12th annual ACM internationalconference on Multimedia*, pages 548–555, 2004.

[4] A.G. Hauptmann, R. Baron, M. Christel, R. Conescu, J. Gao, Q. Jin, W.-H. Lin, J.-Y. Pan, S. M. Stevens, R. Yan, J. Yang, and Y. Zhang. CMU Informedia's TRECVID 200 skirmishes. In Proceedings of the TREC Video Retrieval Evaluation (TRECVID) 2005, Gaithersburg, MD 2005.

[5] A. G. Hauptmann, W.-H. Lin, R. Yan, J. Yang and R. Chen, Extreme Video Retrieval: Joint Maximization of Human and Computer Performance, In ACM Multimedia'06, 2006.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision,* 60, 2 (2004), pp. 91-110.

[7] J. Yang, A. Hauptmann, "Exploring Temporal Consistency for Video Retrieval and Analysis", In Proc. of 8th ACM SIGMM Intl Workshop on Multimedia Information Retrieval (MIR), Oct. 26-27, 2006, Santa Barbara, CA.

[8] M. Christel and R. Conescu, "Mining Novice User Activity with TRECVID Interactive Retrieval Tasks," *Proc. CIVR 2006, Lecture Notes in Computer Science* 4071, Springer, Berlin, 2006, pp. 21-30.

[9] C. Snoek, M. Worring, D. Koelma, and A. Smeulders, "Learned Lexicon-driven Interactive Video Retrieval," *Proc. CIVR 2006, Lecture Notes in Computer Science* 4071, Springer, Berlin, 2006, pp. 11-20.

[10] W. Zhao, Y.G. Jiang, and C.W. Ngo. Keyframe retrieval by keypoints: Can point-to-point matching help? In Conf. on Image and Video Retrieval 2006.