

# Modeling Timing Features in Broadcast News Video Classification\*

Wei-Hao Lin and Alexander Hauptmann  
 Language Technologies Institute  
 School of Computer Science  
 Carnegie Mellon University  
 5000 Forbes Avenue  
 Pittsburgh, PA 15213, U.S.A.  
 {whlin, alex}@cs.cmu.edu

## Abstract

Broadcast news programs are well-structured video, and timing can be a strong predictor for specific types of news reports. However, learning a classifier using timing features may not be an easy task when training data are noisy. In this paper, we approach the problem from the generative model perspective, and approximate the class density in a non-parametric fashion. The results show that timing is a simple but extremely effective feature, and our method can achieve significantly better performance than a discriminative classifier.

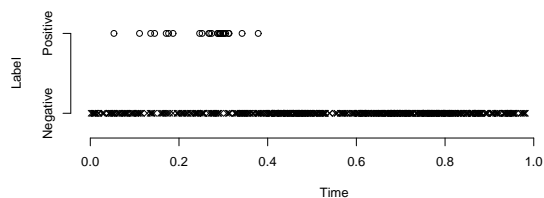
## 1. Introduction

Video classification is arguably the first step toward multimedia content understanding, and has been an active sub-field of multimedia research. A large number of useful features, based on video and audio, that have been proposed for video classification. Human-edited video contains another type of feature in the temporal domain because editors or producers often implicitly or explicitly impose a structure over time; broadcast news programs, for example, are very structured. People watching enough news programs usually can notice that weather reports are not randomly presented in the 30-minute program. Therefore, timing could be an informative feature to distinguish one type of reports from the others.

While it is tempting to apply machine learning technique to acquire the concept without human intervention, the learning task, seeming a very easy at the first

\*This work was supported in part by the Advanced Research and Development Activity (ARDA) under contract number MDA908-00-C-0037.

sight, may not be so obvious after looking closely to the real-world data, shown in Figure 1. While non-weather



**Figure 1. The distribution of the time offset of the positive and negative weather news examples in the TRECVID'03 corpus offset**

news are evenly distributed, it is clear that weather news are not randomly distributed, most of which are centered around time offset 0.3 (corresponding to 10 minutes in a 30-minute program). However, there are many negative, non-weather news stories presenting in the same time period, which confounds learning algorithms trying to find the decision boundary between the two classes. While two classes overlapping heavily in the same region can be due to the incorrect or noisy training labels, the problem of inconsistency and incompleteness of human annotations is so prevalent that any video classification systems must cope with the problem. It will be very difficult for discriminative classifiers to learn here because there is not clear decision boundary to separate two classes of data. Instead, we propose to approach the problem using the generative models. Two modeling approaches will be further described in Section 2. Experiments are conducted to evaluate the effectiveness of the two modeling meth-

ods in Section 3, and concluding remarks are made in Section 4.

## 2. Modeling Timing Features

Statistical classifiers can approach the classification problem either in discriminative models or generative models [4]. Suppose the random variable  $Y$  is the class label, and  $X$  is the one-dimension timing feature. Discriminative models model the posterior probability directly, i.e.  $P(Y|X)$ , while generative models will model the joint probability, i.e.  $P(X, Y)$ . Generally speaking, the performance of the generative models depend heavily on the correctness of the model assumption, while discriminative models are more robust because of fewer assumptions.

We describe how to generate the timing features before describing two modeling approaches.

### 2.1. Timing Feature Representation

Video is often automatically or manually segmented into shots, and video classifiers are asked to make a classification decision at the shot level. A video, therefore,  $\mathcal{D}$  consists of an ordered set of shots  $d_i, i = 1, \dots, |\mathcal{D}|$ . For each shot, the starting and ending offset can be easily obtained, usually in the unit of frames or milliseconds, denoted as  $so(d)$  and  $eo(d)$ , respectively. The middle point of the starting offset and ending offset is used to represent the timing feature  $x_i$  for each shot  $d_i$ .

In a 30-minute news program,  $x_i$  could range from a few milliseconds to millions of milliseconds. Such a large scale may cause numerical problems in classifier training. One simple way to normalize the values in a large range is *linear scaling*, which scales the timing features between zero and one, defined as follows,

$$x_i = \frac{0.5(so(d_i) + eo(d_i)) - \min_k t(d_k)}{\max_k t(d_k) - \min_k t(d_k)} \quad (1)$$

However, linear scaled timing features may be problematic when the length of the video varies much. Suppose most broadcast news programs in a corpus are 30 minutes long, but a few of them are around 20-minutes long. The timing features scaled in the 20-minute range will not be meaningful to those in the 30-minute range. Therefore, instead of dividing by the whole length of the video, we fix the range to 1800000 milliseconds, i.e. 30 minutes, as follows,

$$x_i = \frac{0.5(so(d_i) + eo(d_i))}{1800000} \quad (2)$$

### 2.2. Support Vector Machine

Like all discriminative models, SVM makes assumption on the discriminant functions and use them to classify examples. SVM has been widely used and is very effective in many domains. The basic idea behind SVM is to select a decision hyperplane in the feature space that can separate two classes of data points while keeping the margin as large as possible. The process of finding the hyperplane can be formulated as the following optimization problem,

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2} w^T w + C \sum_{i=1}^l \xi_i \quad (3) \\ \text{subject to} \quad & y_i (w^T \phi(x_i) + b) \geq 1 - \xi_i \\ & \xi_i \geq 0, i = 1, \dots, l \end{aligned}$$

where  $x_i$  is a feature vector,  $i = 1, \dots, l$ ,  $l$  is the size of the training data,  $y_i \in +1, -1$ ,  $y_i$  is +1 when the shot is an positive example, and -1 otherwise,  $\phi$  is the kernel function that maps the feature vector into higher dimension,  $\xi_i$  is the degree of misclassification when the data points fall in the wrong side of the decision boundary, and  $C$  is the penalty parameter that tradeoff between two terms. More details can be found in [1].

### 2.3. Modeling Class Densities

By Bayes' Theorem, the posterior probability  $P(Y|X)$  can be rewritten as the product of the class density  $p(X|Y)$  and the prior class probability  $P(Y)$ , as in the following equation,

$$P(Y|X) = \frac{p(X|Y)P(Y)}{\int_x p(X|Y)P(Y)dx} \quad (4)$$

The key of generative modeling here will be the the class densities  $p(X|Y)$ . A non-parametric density estimation techniques called *kernel density estimation* [5] is chosen to estimate the class density, defined as follows,

$$\hat{p}(X|Y) = \frac{1}{n} \sum_{i=1}^n \frac{1}{h} K \left( \frac{x - x_i}{h} \right) \quad (5)$$

where  $K$  is the kernel function,  $h$  is the bandwidth, and  $n$  are number of examples. We use a Gaussian kernel, and the bandwidth is chosen automatically by the Sheather-Jones selection rule. The reasons for choosing the non-parametric method over parametric Gaussian distributions is that there may be multiple modes in the class densities, which will not be properly modeled by a single-mode distribution like Gaussian distribution.

### 3. Experiments

#### 3.1. Testbed, Classification Tasks, and Evaluation Metric

We choose the video corpus of TRECVID 2003 [3] as the testbed in this paper. The corpus consist of broadcast news programs, including ABC, CNN, and C-SPAN news programs, and we can compare our results with participants in TRECVID because it has been an open contest of content-based video retrieval systems held by NIST since 2001. Among 17 video classification tasks in TRECVID 2003, two tasks, Sporting Events and Weather News, were hypothesized to contain strong timing cues, and chosen here to evaluate the effectiveness of different timing modeling techniques. The official definitions of these two tasks are listed as follows,

**Sporting Event** shot contains video of one or more organized sporting events

**Weather News** shot reports on the weather

The basic statistics of these two tasks in the training and testing <sup>1</sup> set are listed in Table 1. C-SPAN data are not included because they contained no sporting events or weather news shots. Note that positive examples are very rare in the training data, around 1% in both tasks, which resulted in classifiers having difficulty modeling the concept.

Set	Source	Task	Positive	Total
Training	ABC	Sporting Event	303	25630
		Weather News	71	
	CNN	Sporting Event	303	21696
		Weather News	215	
Testing	ABC	Sporting Event	26	16593
		Weather News	7	
	CNN	Sporting Event	559	15282
		Weather News	159	

**Table 1. Basic statistics of two video classification tasks in the TRECVID 2003 video corpus**

TRECVID uses MAP (Mean Average Precision) as the evaluation metric, and we use the same metric in order to fairly compare with the TRECVID submissions in 2003. AP (Average Precision) of a rank list  $\mathcal{A}$

<sup>1</sup>The number of the positive examples in the testing set is underestimated because TREC used the pooling method to evaluate the rank lists submitted by participants. Only video shots that are officially evaluated as positive are counted here

is defined as:

$$AP(\mathcal{A}) = \frac{1}{|\mathcal{A}^+|} \sum_{d \in \mathcal{A}^+} \frac{U^+(d) + 1}{U(d) + 1} \quad (6)$$

where  $\mathcal{A}^+$  is a set of all positive examples in  $\mathcal{A}$ ,  $U(d)$  is a function returning the number of examples ranked higher than the example  $d$  in  $\mathcal{A}$ , and  $U^+(d)$  is a function returning only the number of positive examples ranked higher than  $d$ . MAP takes the average of AP over a set of queries or tasks.

#### 3.2. Modeling Class Densities

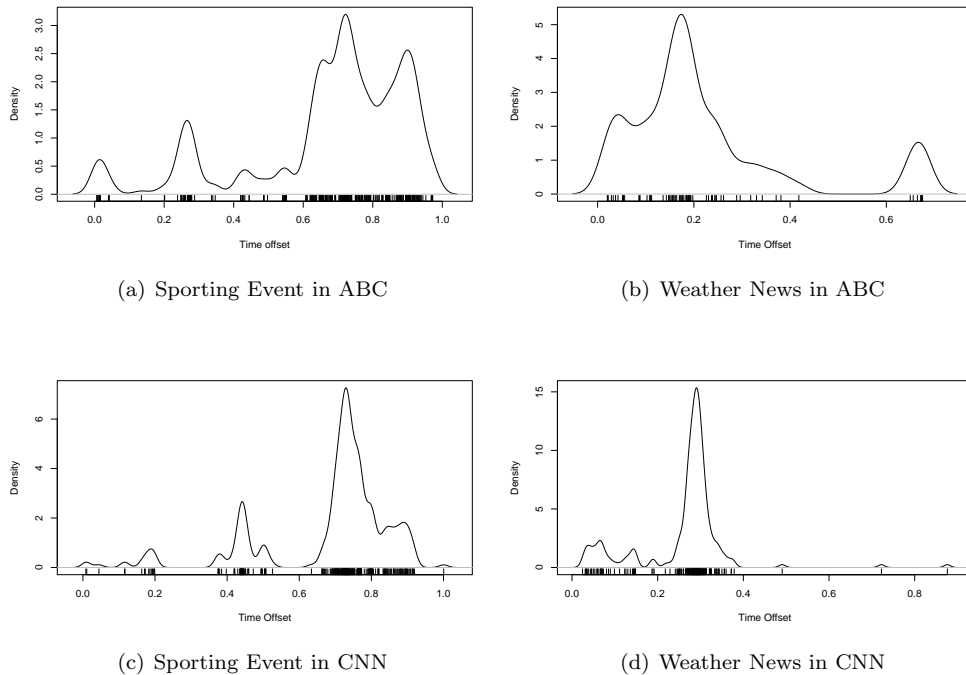
The class densities  $p(X|Y)$  estimated by kernel density estimation are plotted in Figure 2. Clearly the class densities are multi-modal distributions, which justifies our choice of a non-parametric estimation method. The estimated densities meet our expectations that weather news is usually presented in the first second of the news program, while sports news in second third of the program. However, the class densities are not totally the same, which implies we should model each channel separately instead of simply ignoring the news channel characteristics.

#### 3.3. Classification Results

Methods	Sporting Event	Weather News
SVM (Full Training)	0.024	0.009
Kernel Density (Full Training)	<b>0.131</b>	<b>0.567</b>
SVM (Separate Training)	0.005	0.002
Kernel Density (Separate Training)	<b>0.220</b>	<b>0.807</b>
Random Baseline	0.022	0.008
TRECVID'03 median	0.152	0.417
TRECVID'03 best	0.708	0.856

**Table 2. The experiment results of the classification tasks**

The classification results using the timing features are shown in Table 2. The *full training* condition means we trained a classifier for the whole data collection, while in *separate training* we trained a classifier for each individual news channel, and merged the ranked lists using logistic regression as a global mapping function, described in [2]. The results strongly favor the generative model, and SVM breaks down and performs close to the random baseline. As described in previous section, it is very hard to do discriminative training like



**Figure 2. The estimated class densities  $p(X|Y)$**

SVM here when data are noisy or incomplete. Moreover, the performance of separate training runs is significantly better than that of full training, which is not surprising because, as shown in Figure 2, each news channel has very different timing profiles. Building a separate classifier for each individual news channel can capture the idiosyncrasies of each news channel, while full training totally ignores specific source characteristics at the expense of the classification performance.

The performance of merging results from individual classifiers using kernel density outperforms the median performance of the TRECVID 2003 participants. While timing features can be extracted without any effort, it appears that timing features are still largely ignored, or cannot be easily leveraged because of the difficulty in discriminative learning.

## 4. Conclusions

In well-structured video like broadcast news programs, timing features can provide strong cues for classifying specific types of video, but need to be carefully modeled. By modeling the class density in a non-parametric fashion, generative models is shown here to significantly outperform discriminative models when labeled data are incomplete and noisy.

## References

- [1] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge University Press, 2000.
- [2] W.-H. Lin and A. Hauptmann. Merging rank lists from multiple sources in video classification. In *Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME)*, Taipei, Taiwan, June 27-30 2004.
- [3] NIST. Guidelines for the TRECVID 2003 evaluation. Webpage, 2003. <http://www-nlpir.nist.gov/projects/tv2003/tv2003.html>.
- [4] Y. D. Rubinstein and T. Hastie. Discriminative vs informative learning. In *Proceeding of the Third International Conference on Knowledge Discovery and Data Mining (KDD)*, Newport Beach, CA, U.S.A., August 14-17 1997.
- [5] D. W. Scott. *Multivariate Density Estimation: Theory, Practice, and Visualization*. Wiley, New York, 1992.