

# Naming Every Individual in News Video Monologues

Jun Yang

Alexander G. Hauptmann

School of Computer Science  
Carnegie Mellon University  
5000 Forbes Ave., Pittsburgh, PA 15213, USA  
1-412-268-(9747, 1448)  
{juny, alex}@cs.cmu.edu

## ABSTRACT

Naming every individual person appearing in broadcast news videos with names detected from the video transcript leads to better access of the news video content. In this paper, we approach this challenging problem with a statistical learning method. Two categories of information extracted from multiple video modalities have been explored, namely *features*, which help distinguish the true name of every person, as well as *constraints*, which reveal the relationships among the names of different persons. The person-naming problem is formulated into a learning framework which predicts the most likely name for each person based on the features, and refines the predictions using the constraints. Experiments conducted on ABC World News Tonight and CNN Headline News videos demonstrate that this approach outperforms a non-learning alternative by a large amount.

## Categories and Subject Descriptors

H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing -*indexing methods*

## General Terms

Algorithms, Experimentations

## Keywords

Person naming, multi-modality, machine learning, equivalence constraint, broadcast news video

## 1. INTRODUCTION

Detecting people appearing in digital videos and recognizing their identity is essential to better indexing and access of video content. This is particularly true for broadcast news video, whose semantics are mainly concerned with the activities of various people as news subjects. Compared with face detection [10], identifying people's names and roles in news videos is of greater interest to users since it allows intelligent access to the video content, such as finding all the events involving a named person, or probing the name and identity of an unknown face.

Associating people appearing in news videos with their names consists of two inverse sub-problems: (1) *What are the video segments where a named person appears?* (2) *What is the name of a person appearing in a given video segment?* The first problem,

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.

MM'04, October 10–16, 2004, New York, New York, USA.  
Copyright 2004 ACM 1-58113-893-8/04/0010...\$5.00.

referred to as named person finding, has been relatively well studied [13] and evaluated at TREC Video Retrieval Evaluation (TRECVID) [12]. The second one, virtually a person naming problem, is more difficult and interesting research-wise since it attempts to associate non-textual data with textual data instead of the other way around (as in the first problem). However, only limited efforts have been devoted to this problem [6][9]. Note that person naming is fundamentally a different problem from face recognition [16] because instead of matching against a limited number of known faces in the database, it needs to label potentially a huge number of people in videos, most of which have never appeared before. Moreover, the highly heterogeneous settings on lighting and face pose/size in news videos make face recognition unlikely to produce good results useful to name the people. This paper attempts to solve person-naming problem by a statistical learning approach without using facial information.

In essence, person naming can be viewed as a classification problem of matching a visible person with one of the names extracted from the local video transcript. It is not immediately obvious what features are useful for this task. In this paper, we examine a variety of potentially useful evidences derived from multiple modalities, and divide them into two groups, namely *features*, which help distinguish the correct name of each person, and *equivalence constraints*, which reveal the relationships among the names of different people. An SVM learning-based model is formulated and trained to predict the most likely name for each person based on the features, and the constraints are implemented as "score aggregates" to refine the predictions on the people involved in the constraints. The proposed approach is evaluated on the TRECVID dataset with encouraging results obtained.

Our approach consists of several steps. First, for a set of people to be named, we collect their candidate names by named-entity detection (Section 3). Then, a variety of features and constraints are extracted from multiple video modalities to capture the properties of the people and names (Section 4). Finally, our learning model is trained to predict for each person the most likely name based on the features and the constraints (Section 5). Experiments are conducted to show the effectiveness of our approach (Section 6).

## 2. RELATED WORK

On the surface the person-naming problem is a variation of the well-studied face recognition problem [16]. The goal of face recognition is to determine the identity of an unknown face in still or video images from a database of known faces [16]. It involves detecting the face(s) from a given image, extracting features from face regions, and determining their identities. It seems that person naming can be done by face recognition if every face in the database is associated with a name. However, there is a fundamental difference between the two problems: while face recognition only deals with a finite number of distinct faces stored in the database,

person naming needs to label virtually an infinite number of distinct people that will possibly appear in videos. Particularly for news videos where people as news subjects constantly change with time, building a database containing *all* the distinct faces is simply impossible. There is also a practical difficulty of applying face recognition to person naming. Existing face recognition methods work effectively for frontal faces on mug shots under homogeneous lighting conditions, but are unlikely to produce useful results on news videos where people appear with very different poses, sizes, and lighting conditions. Due to these two difficulties, face recognition is not used in our approach.

A semi-automatic method is proposed in [15] to name face images in family albums. It starts with manually naming faces in some photos and then propagates the labeled names to other photos based on facial similarity and image similarity under certain temporal constraints. Though it offers a step beyond face recognition by including image similarity, this work still deals with a "finite identity matching" problem and thus not for our task.

The Name-It [9] system has been proposed to associate names and faces in news videos based on the co-occurrence between the detected faces and names extracted from the transcript. A face image is labeled with the name that has the largest temporal overlap with a group of images containing faces similar to the given one. The face images corresponding to a given name can be found in a similar way. Though theoretically sound, the robustness of this method is affected by the unreliable eigenface-based face similarity in the heterogeneous settings of news videos, as well as the temporal misalignments between names and faces. No serious performance evaluation has been reported on this work. A similar work [2] labels face images in online news articles with names extracted from news captions by exploring co-occurrence between the clustered faces and the names, which obtains good results. However, the fact that a person's face images in news videos are of much lower quality and homogeneity than those in textual news articles (where many are "standard" frontal faces) makes face clustering unlikely to be effective in our task. Nevertheless, our approach still utilizes the co-occurrence information, derived from the more reliable image similarity and speaker identities.

Named Faces system [6] built a database of named faces that allows users to query a person's name by submitting an image containing the target face. The database is populated with the faces from news videos whose names are overlaid on video frames. These names are first recognized using video optical character recognition (video OCR) [8], and the inaccurate recognition results are corrected using a dictionary of names by fuzzy match. Unfortunately, this work does not label the large number of people whose names appear not in overlaid text but only in the closed-captions. Since these people's names are not visible from the video frame, labeling their names is of greater research importance, which is also the motivation of this paper.

### 3. EXTRACTING CANDIDATE NAMES

To make the problem more tractable, two simplifications are made to the scope of the people to be labeled. First, only the people whose names appear in the video transcript of the corresponding news stories are labeled. A news story consists of a sequence of consecutive camera shots on a certain news event [13], and the video transcript is obtained from either the closed-captions or automatic speech transcription of a video. Our study excludes anonymous people (e.g., unknown interviewees on the street) and

the people whose names appear only in overlaid text on video frames, since recognizing these names would rely exclusively on improvements of the state-of-the-art video OCR techniques [8]. As a second simplification, we only label the person who is giving a speech individually in a video shot, i.e., an individual performing a 'monologue'. This rules out the cases where more than one people are speaking in a shot, but NOT the cases where a person is speaking in the foreground with some silent faces on the background. Since most news subjects give an 'monologue' speech somewhere in a news story, this simplification will not skip many important people. Moreover, multiple talking people co-existing in a single shot are extremely hard to distinguish without reliable face recognition, which, however, is beyond the state of the art in news video footage. Therefore, this case is not addressed in our current study, though it is an interesting future topic. With the two simplifications, the person-naming problem boils down to labeling monologue shots with one of the names in the transcript of corresponding news stories. Note that monologue shots include those of anchors, reporters, and news-subjects, since all are likely to give a monologue speech in news videos.

The candidate names for a monologue shot are the names appearing in the chunk of transcript that corresponds to the new story containing this shot. To obtain the candidate names automatically, we apply the named entity extraction to identify proper names like people, organization, and location names from the video transcript. The extraction approach is similar to that used in the BBN named entity detection system [1]. First, a corpus of broadcast news video transcript is manually labeled with named-entity tags on every word, and a language model is trained from the sequences of tag-word pairs. To detect named entities from unlabeled transcript, we build a Hidden Markov Model (HMM) that models each type of name-entity tag as a state and words as the observations at each state. A Viterbi algorithm is used to find the most likely sequence of states that generates the observed sequence of words in the transcript. The words tagged as "*person*" are people's names. Since the transcript is temporally aligned with the video, the names falling into the boundary of each new story are identified as candidate names of the shots in that story.

## 4. FEATURES AND CONSTRAINTS

### 4.1 Features

As mentioned in Section 1, *features* are the information that directly helps distinguish the correct name of a given shot from the other candidate names. The multi-modal nature of digital news video enables features extracted from multiple evidences to complement each other. Moreover, the state-of-the-art video analysis techniques have equipped us with a variety of semantic-level features that can be computed from their low-level counterparts with a good degree of confidence, such as shot and story boundaries, speaker identification, etc. They serve as our building blocks towards more sophisticated tasks such as person naming.

#### 4.1.1 Transcript Clues

Each type of broadcast news has certain transcript clues which, if available, allow us to identify reliably the anchor and reporter shots as well as anchor and reporter names. For example, an anchor of ABC World News Tonight normally uses one of a few fixed ways to end the broadcasting, such as "*Im Peter Jennings. Have a good night*", from which we can tell that "*Peter Jennings*" is the name of an anchor and the person in the shot is the anchor. Also, an anchor often introduces a reporter in a certain way, such

as "ABC's Linda Douglass has the story", indicating that this is an anchor shot and "Linda Douglass" is a reporter's name. As there are only a small number of clues, we manually model them as "templates" and find all their occurrences in the transcript automatically. Since anchor and reporter names are highly recurrent in the broadcastings of many days, they are perfectly identified by cross-verifying a large number of clues in our preliminary experiment on the test data. Once the anchors' and reporters' names are known, the rest are the names of news subjects. We will see that knowing the type of each name is very important to our task.

#### 4.1.2 Video OCR Text

As exemplified in Figure 1, when a person is speaking, his/her name is often displayed on the screen as overlaid text, which, if precisely recognized, tells the correct name of the person. However, the video OCR technique [8] we used to process the overlaid text produces highly erroneous results on news video due to its compression loss and low resolution. For example, in Figure 1 the name *Newt Gingrich* is recognized as "nev~j ginuhicij".

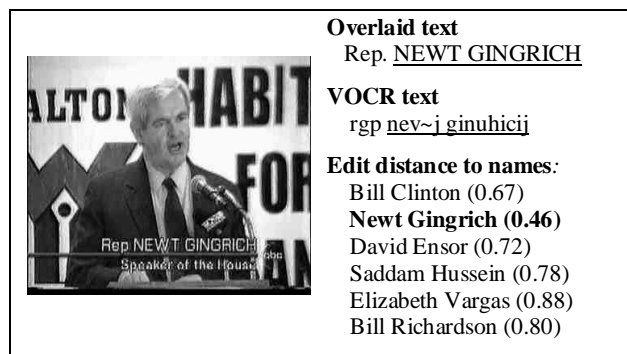


Figure 1. An example of overlaid text with its video OCR

Despite its poor quality, the VOOCR text still provides weak clues pointing to the right name of the speaker, based on our observation that it usually looks more similar to the right name than to any other name. A similarity measure between two text strings is needed to tell how similar the VOOCR text is to each candidate name. The edit distance[6], calculated as the number of editing operations (insertion, deletion, and substitution) needed to convert one string into another, is used in our method. However, since names are often mixed up with other information (e.g., title, affiliation) in the overlaid text, it is hard to tell which portion of the VOOCR text corresponds to a name. To overcome this problem, a sliding window is used to find the portion of the VOOCR text that maximally matches with a candidate name, and the corresponding (minimal) edit distance normalized by the length of the name is used as the distance between the VOOCR text and the name. Figure 1 shows the normalized edit distance between each candidate name and the VOOCR text, where the correct name (*Newt Gingrich*) has the smallest distance as expected.

#### 4.1.3 Speaker Identity

The speech accompanying the news video is segmented and the segments are clustered using LIMSI speech detection and recognition system [4]. Presumably, the speech segments in each cluster belong to the same speaker and they are assigned a unique identity (speaker ID). Since a shot may temporally overlap with multiple speech segments with different speaker IDs, the ID with the maximal temporal coverage in the shot is regarded as the ID of the primary speaker of the shot. Several features are derived from the

primary speaker ID of a shot, including (a) whether it occurs in more than one story, (b) whether the speaker of this ID utters a specific candidate name, (c) whether it temporally dominates the story (i.e., having maximal coverage), and (d) the number of neighboring shots where the same ID is the primary speaker ID (denoted as "span"). Figure 2 shows the speaker IDs corresponding to a shot sequence with the "span" of each monologue shot.

Feature (a) helps identify anchor shots since only the anchor's speaker ID may go across story boundaries. Feature (b) makes sense because news subjects rarely say their own names, while anchors and reporters sometimes do. So if a name is uttered by the speaker of a given shot, it cannot be the name of the speaker unless it is an anchor or reporter's name. Feature (c) and (d) are useful in identifying news subjects since their speaker ID seldom temporarily dominates a story or spans across multiple shots. Note that there are exceptions in some monologue shots where the speaker's voice is replaced or overwhelmed by a "background voice" from the anchor or reporter. Although the speaker identity features are misleading in such cases, they are too rare to deserve particular handling.

#### 4.1.4 Name Positions

This class of features investigates the temporal position of a name in relation to the shot in question. From our observation, the true name of the person in a shot is most likely to be mentioned before the shot, sometimes after it, but rarely within the shot unless the person is an anchor or a reporter. It is therefore useful to know whether a candidate name appears before, within, or after a shot, and if not within it, how close they are. Moreover, if there are multiple names before or after the shot, in which order a name appears in the sequence of names is also important.

#### 4.1.5 Temporal Structure

Broadcast news footage has a relatively fixed structure. A typical news story is first briefed by an anchor, followed by several shots showing the news event where news-subject(s) and reporter(s) may appear, and usually concluded by the reporter or the anchor. Although there are exceptions, such as some short news stories that contain only one anchor shot, this structure helps identify the type of a person and thus increases the chance of naming him/her correctly. We examine the position of a given shot in the shot sequence of the corresponding news story, and use its offset from the start and end of the story as two features, plus an additional feature giving the number of shots in the story.

## 4.2 Constraints

Unlike features, constraints suggest the *equivalence* among the names of multiple shots rather than the names of the shots. Hence, they are not useful when predicting the name of each individual shot. But when a set of shots bounded by a constraint are being labeled, chances are that the constraint may improve the prediction of one shot from that of another shot whose name label is predicted with higher confidence, and boost the overall accuracy of predictions. Two types of constraints are discussed below.

#### 4.2.1 Constraints based on image similarity

Based on our observation, if a news-subject person appears several times in a news story, the shots with his/her appearance are likely to be of similar settings including background, clothes, illuminations, and camera position, resulting in high visual resemblance among these shots. Figure 2 shows such an example, where a military analyst (shot 2, 4, 9, and 11) is discussing the Iraq issue

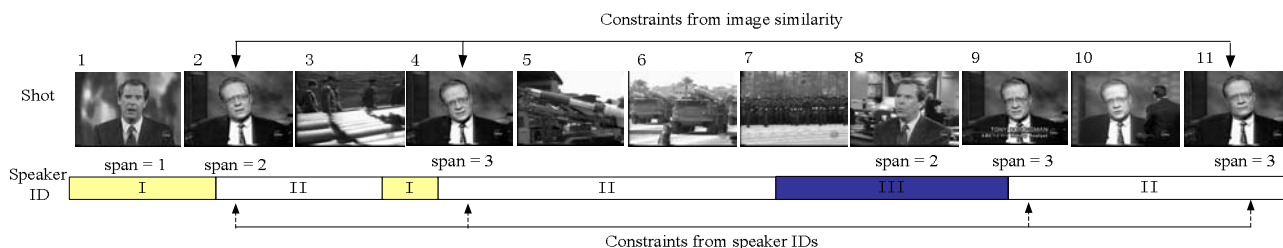


Figure 2: A sequence of shots in a new story with constraints derived from image similarity and speaker ID

with the anchor (shot 1, 8). As we can see, the military analyst stays in a fixed setting throughout the whole news story, and the shots in which he appears are indeed very similar. If there is a constraint on these shots, the name label of one shot (say, shot 9, which is probably better predicted since it has VOOCR text) can improve the labels of the other 3 shots.

Therefore, if the key-frames of a set of shots are visually similar to each other, we create a constraint to enforce the same name label on these shots. Here the key-frame similarity is measured by Euclidean distance on RGB color histograms computed from  $5 \times 5$  rectangular image regions, which preserve not only visual but also structural information, and a threshold is set empirically to determine how similar two key-frames need to be for a constraint to be created. The solid line in Figure 2 indicates a constraint derived from image similarity. Note that the reverse conclusion, i.e., dissimilar shots contain different persons, is not always true, such as shot 1 and 8 in Figure 2. Moreover, one cannot go across the story boundaries to derive such constraints, since a person does not necessarily appear in the same setting in different news stories.

#### 4.2.2 Constraints from Speaker ID

With few exceptions, the speaker ID of a monologue shot belong to the speaker in the shot. Thus, shots with identical speaker ID should contain the same person, leading to a constraint enforcing identical name label on these shots. In Figure 2 the speech segments of the same speaker ID are labeled with the same color, and we can see that the primary speaker ID of shot 2, 4, 9, and 11 are identical, among which a constraint is created as indicated by the dash line. Unlike its counterpart on image similarity, a constraint derived from speaker ID may consist of shots from different news stories, because the same speaker ID from multiple stories still indicate the equivalence of speakers. Again, the reverse conclusion (i.e., shots with different speaker IDs contain different persons) is not true here, since the LIMSI speaker identification engine is prone to label the speeches of the same person with different IDs. For instance, shot 1 and 8 are labeled with different speaker ID though they have the same person.

Between the two types of constraints, we expect those derived from image similarity to be more effective than those from speaker IDs, since image similarity provides *non-redundant* information not overlapped with the features in Section 4.1, while speaker ID does not. However, speaker ID leads to *global* constraints across story boundaries while image similarity only produces *local* constraints. Despite the differences, the two types of constraints have many overlaps, e.g., the constraints on shot 2, 4, and 11 in Figure 2 can be derived from both sources. In practice, we can let these two sets of constraints to complement each other (using their union) or cross-validate each other (using their intersection). These possibilities will be explored in Section 6.

## 5. LEARNING TO NAME PERSONS

The multitude and diversity of features makes the person-naming task unlikely to be successful using *ad-hoc, non-learning* approaches. Moreover, the problem formulated as choosing a shot's name from a set of candidate names makes many classification methods applicable, where the classifiers can be learned from training data to make sophisticated and justified (naming) choices based on a variety of features. This section discusses how to formulate and train a learning model for our task and how to extend it with constraints.

### 5.1 Problem Formulation

An intuitive formulation of the person naming problem is to learn a mapping function  $f: S \rightarrow N$ , where  $S$  denotes the set of monologue shots that contain the people to be labeled, and  $N$  denotes a finite set of candidate names as class labels. A learning algorithm takes a set of training examples  $(s_1, n_1), \dots, (s_m, n_m)$  as input, where  $n_i \in N$  is the name manually labeled on shot  $s_i \in S$ , and produces an optimized mapping function  $f$  to predict the name of unlabeled shots. However, this straightforward formulation has many practical difficulties: (1) The size of  $N$  is extremely large since there exists a huge number of names in news videos, which makes the classifier *unlearnable* from the relatively limited training examples. (2) Despite how large  $N$  is, new names not included in  $N$  will inevitably emerge in news videos, and these names will never be predicted under this formulation. (3) The candidate names of each shot are those appearing in the story containing the shot, which is only a small subset of  $N$ . This constraint is not embodied in the current formulation. (4) Many features suggested in Section 4.1 are associated with names, and the same name appearing at different places carries with different features. This contradicts with the fact that  $f$  is a function of only  $s_i$  but not  $n_i$ .

Given the above difficulties, we suggest an alternative formulation, which, instead of directly predicting the name of each shot, predicts how likely a monologue shot matches a candidate name. This formulation has a different mapping function  $g: S \times N \rightarrow R$ , where  $S \times N$  is a set of pairs  $\langle s_i, n_j \rangle$  between a shot  $s_i \in S$  and one of its candidate names  $n_j \in N$ . Note that  $n_j$  must appear in the same news story as  $s_i$  to be a valid candidate name.  $R \in [-1, 1]$  is a real value indicating the degree of association between  $n_j$  and  $s_i$ , with "1" denoting match and "-1" denoting mismatch.

The training examples  $(s_1, n_1), \dots, (s_m, n_m)$  need to be restructured to fit into the new mapping function  $g$ . Specifically, we rewrite each training example  $(s_i, n_i)$  (where  $n_i$  is the true name of shot  $s_i$ ) as a series of tuples like:

$$(\langle s_i, n_{i1} \rangle, r_{i1}), \dots, (\langle s_i, n_{ij} \rangle, r_{ij}), \dots, (\langle s_i, n_{iK_i} \rangle, r_{iK_i})$$

$$r_{ij} = \begin{cases} -1 & n_{ij} \neq n_i \\ 1 & n_{ij} = n_i \end{cases}$$

where  $\{n_{ij}\}_{j=1, \dots, K_i}$  is a set of  $K_i$  candidate names of  $s_i$ , and  $r_{ij}$  is the score showing whether or not  $n_{ij}$  is the correct name of  $s_i$ . Note that the number of training data is much larger since one training example in the previous formulation translates to  $K_i$  examples here. Similarly, a shot  $s_i$  to be labeled is first mapped to a set of tuples as  $\{\langle s_i, n_{ij} \rangle\}_{j=1, \dots, K_i}$ , which are fed into the trained model to produce a set of scores  $\{r_{ij}\}_{j=1, \dots, K_i}$ . The name predicted for  $s_i$  is the one with the highest score  $n_i = \arg \max_{n_{ij}} \{r_{ij}\}_{j=1, \dots, K_i}$ .

Note that in this model each example  $\langle s_i, n_{ij} \rangle$  actually refers to a vector of the features of shot  $s_i$  and name  $n_{ij}$  (cf. Section 4.1).

By transforming a multi-class classification problem into what is essentially a regression problem (or a binary classification problem once a threshold is set), we circumvent all the modeling difficulties raised before. First, the mapping function  $g$  can be reliably learned with sufficient training data. New names emerged will be added to the data space (instead of to the class labels) and thus can be predicted. Furthermore, for each shot  $s_i$  we only add the pair  $\langle s_i, n_{ij} \rangle$  such that  $n_{ij}$  is a valid candidate of  $s_i$ . Finally, since  $g$  is a function of both the shot and the name, all the features in Section 4.1 can be represented.

## 5.2 Learning with Support Vector Machines

Support vector machine (SVM) [3] is a general machine learning algorithm with some nice properties that make it a suitable choice for our task. Specifically, we have a large number of correlated features, while SVM is capable of handling mutually dependent, high-dimensional features. The decision function of SVM is as:

$$y = \text{sign}(f(x)) \text{ and } f(x) = \sum_{i=1}^N y_i a_i K(x, x_i) - b$$

where  $x$  is an input feature vector,  $f(x)$  is its decision margin that is converted into a class label  $y \in \{-1, 1\}$  through a sign function.  $x_i$  represents the  $i^{\text{th}}$  of the  $N$  training examples,  $\{a_1, \dots, a_N\}$  and  $b$  are the parameters of the model to be learned, and  $K(x, x_i)$  is a kernel function. In our problem setting,  $x$  is the features extracted from a shot-name pair  $\langle s_i, n_{ij} \rangle$ . Since a real-value score is needed to indicate the degree of association between each shot-name pair, we set the score  $r_{ij}$  equal to  $f(x)$ . The radial basis function (RBF) is used as the kernel function due to its good performance.

## 5.3 Integrating the Constraints

The model above uses features but not constraints. To integrate the constraints into it, we add another level on top of the SVM to aggregate the scores predicted for a candidate name with different shots involved in the constraints. Suppose a constraint  $c^t \in C$  is expressed as:

$$C^t = (s_1^t, \dots, s_i^t, \dots, s_{M^t}^t)$$

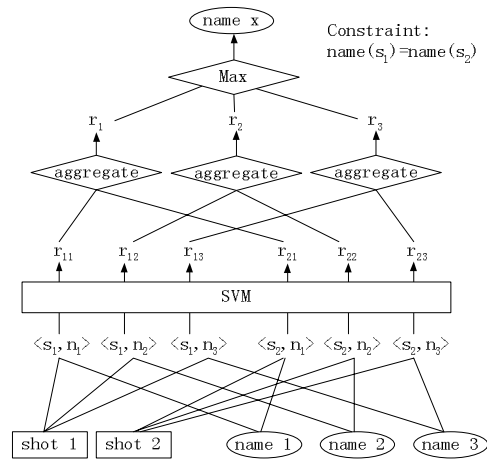


Figure 3: Labeling two shots with a constraint

which indicates that the true name of shot  $s_1^t$  through  $s_{M^t}^t$  are identical. Each shot  $s_i^t$  in the constraint has a set of candidate names  $N_i^t = \{n_{ij}^t\}_{j=1, \dots, K_i^t}$ , which can be different from each other since these shots do not necessarily come from the same story. If the constraint is correct, the true name must appear in every candidate set. Therefore, the actual candidate set of these shots is the intersection of their respective candidate sets, denoted as  $N^t = \{n_j^t\}_{j=1, \dots, K^t}$ . Our goal is to compute the overall score  $r_j^t$  for a name  $n_j^t$  by aggregating its scores  $\{r_{ij}^t\}_{i=1, \dots, M^t}$  with each shot  $s_i^t \in C^t$ . There are different ways to aggregate the scores, such as maximum-aggregate, which sets the overall score to the maximum score in  $\{r_{ij}^t\}$ , or average-aggregate, which computes the average score, given as:

$$r_j^t = \max (r_{ij}^t)_{i=1, \dots, M^t} \text{ and } r_j^t = \frac{1}{M^t} \sum_{i=0}^{M^t} r_{ij}^t$$

Finally, the name with the highest aggregated score is assigned to all the shots in  $C^t$ . The whole process for labeling two shots bound by a constraint is illustrated in Figure 3. The success of the aggregation relies on the assumption that the correct name receives a high score on at least one shot in the constraint (for maximum-aggregate), or its average score over these shots is the highest (for average-aggregate). Both are not always true, and if the assumption does not hold for a particular constraint, using that constraint may degrade the performance rather than help it.

## 6. EXPERIMENTS

### 6.1 Dataset

The dataset used for training and testing our person naming approach was derived from the development set of TRECVID 2004 [11]. It contains ABC World News Tonight broadcasting in 10 random days within a period of 3 months (Feb. to Apr. 1998), with the shot and news story boundaries provided by TRECVID. The monologue shots to be labeled are first detected by our monologue detector [11], which is an SVM classifier built on features of speeches, face detections, video OCR, etc. This detector is the best-performing one on the monologue detection task in TREC-

VID 2004, where it achieves over 80% average precision. We manually inspected its detection results to eliminate false alarms and add misses. This results in a total of 498 monologue shots, among which 122 shots contain anonymous people or people whose names only appear in the text overlaid on video frames. Since currently our method cannot deal with such people, only the remaining 376 people are used for testing. The people names are extracted from the closed-captions accompanying the video, which is temporally aligned by matching them with the automatic speech transcripts. Thus, the candidate names of a shot are those that fall into the boundary of the corresponding story, plus the name of the anchor of that day's broadcasting, which does not usually appear in every news story. The true names of all the test shots were manually labeled, which took about 35 person-hours. The time needed for collecting truth is the primary difficulty that prevents us from running experiment on a larger dataset such as the whole TRECVID data. In comparison, training the classifier and processing the test data is extremely efficient (each less than 30 seconds). The features extracted for each shot-name pair are summarized in the appendix, which are linearly normalized into [0,1] to avoid the bias towards features with large value domains.



Figure 4: Examples of news subjects, anchors, and reporters

As mentioned, the monologue shots include those of the anchors, reporters, and news subjects. Although users are mainly interested in knowing the names of the news subjects, we include the other two types of people since their appearances are temporally intermingled with those of news subjects and distinguishing them merely from visual clues is much harder than expected. Figure 4 shows examples of the three types of people, which look similar to each other, especially between news-subjects and reporters. This implies the possibility that a user asks for the name of an anchor or a reporter without knowing it is not a news subject. Moreover, we have built the person naming approach into the client of



Figure 5: Interface showing the people labeled with names

our Informedia digital video library system [5]. Figure 5 shows a pane of the client displaying the shots of a given video segment, where a bounding box is drawn around each detected face. As we can see, when the mouse is moved into the bounding box of a face, a pop-up window appears to show the two most likely names predicted for this person as well as the types of the person.

## 6.2 A Non-learning Baseline

No serious evaluation has been reported on perhaps the only existing system for person naming, Name-It [9], and its complexity makes it infeasible to duplicate its algorithm in our experiment. To provide a performance benchmark, we have implemented a non-learning approach for person naming as a baseline. As shown in Figure 6, this approach works in a rule-based manner: it first distinguishes the type of a shot being examined as anchor, reporter, or news-subject based on some simple evidences including transcript clues and its temporal position; then, it assigns names of the predicted type in the story to the shot in the ascending order of temporal distance (from the name) to the shot. This baseline provides a good benchmark showing how much a learning-based model can outperform a non-learning alternative.

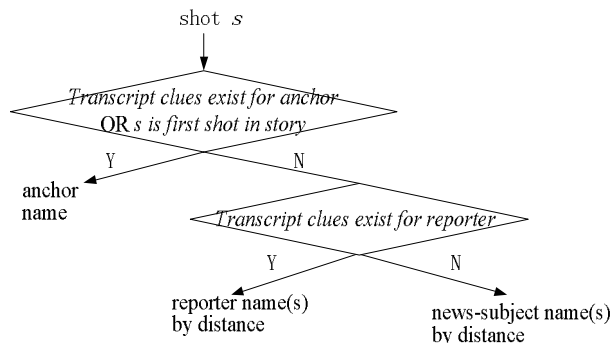


Figure 6: The algorithm of a non-learning baseline approach

## 6.3 Accuracy of Person Naming

Our experiments are conducted using 10-fold cross-validation. That is, each time we train our model using 9 out of 10 days' news videos and test it on the video of the remaining day. This is repeated 10 times, each time using the video of a different day for testing. Four approaches are evaluated, namely the baseline approach (**Baseline**), our learning model without using constraints (**LM**), with constraints implemented as maximum-aggregate (**LM+MaxCons**) or as average-aggregate (**LM+AvgCons**). The constraints used here are the intersection of those derived from image similarity and those from speaker ID. Each approach returns for each shot a list of names ranked by the descending possibility of being the correct name. The performance is evaluated by the naming accuracy, which is the ratio of the shots whose correct name is returned within the top- $n$  names. We average the naming accuracy over the 10 runs of the cross validation. Table 2 reports the average naming accuracy of the four approaches with  $n$  varying from 1 to 3. Besides the overall accuracy, we also report the accuracy on different categories of people, i.e., anchor, reporter, and news subject, among which the accuracy on news-subject is of the greatest interest. The number of shots in each category is also shown in Table 1.

In the experiment, the 376 testing shots constitute totally 1852 shot-name pairs, with 4.9 candidate names per shot. This means

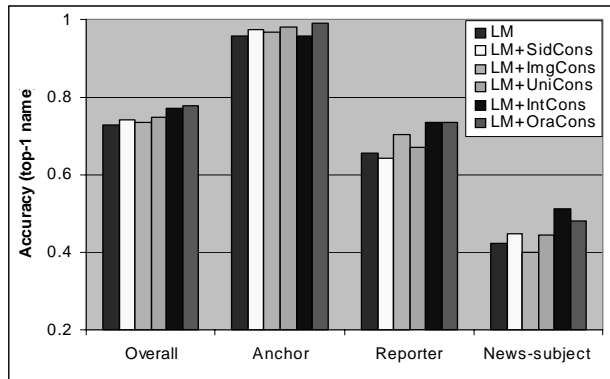
**Table 1: Naming accuracy of different approaches**

		Overall (376)	Anchor (187)	Reporter (64)	News subject (125)
Top-1 name	Baseline	0.561	0.834	0.359	0.256
	LM	0.728	0.958	0.656	0.424
	LM+MaxCons	0.763	0.957	0.703	0.504
	LM+AvgCons	0.771	0.957	0.734	0.512
Top-2 name	Baseline	0.659	0.860	0.422	0.48
	LM	0.853	0.973	0.859	0.672
	LM+MaxCons	0.867	0.979	0.875	0.696
	LM+AvgCons	0.856	0.979	0.875	0.664
Top-3 name	Baseline	0.710	0.877	0.515	0.56
	LM	0.896	0.983	0.922	0.752
	LM+MaxCons	0.880	0.978	0.922	0.712
	LM+AvgCons	0.875	0.978	0.922	0.696

the chance that a shot is correctly named by random guess is about 20% on average, or 40% and 60% if the second or third likely name can be guessed. Given this "random baseline", our **Baseline** approach achieves a reasonably good overall accuracy (56.1%) at top-1 name, but its accuracy on news-subject (25.6%) is only slightly better than chance. When the top-2 and top-3 names are checked, however, the advantage of **Baseline** shrinks particularly on reporters and news-subjects. **LM** significantly outperforms **Baseline** on the accuracy of every category, no matter whether the top-1, 2 or 3 names are checked. It does almost a perfect job in naming anchors, owing to the variety of features capable of distinguishing anchor shots. Its accuracy on reporter is almost the double of that of **Baseline**, and its accuracy on news-subject is about 50% higher than **Baseline**. This verifies the effectiveness of our learning-based method, though it has much room for improvement as its accuracy on news-subjects is less than 50%.

As shown in Table 1, both **LM+MaxCons** and **LM+AvgCons** outperform **LM** at top-1 name, indicating the effectiveness of using constraints in refining the naming results. Looking into the accuracy on different categories, we notice that the improvement is mainly from news-subject, which makes the improvement more valuable given the importance of news-subjects. Tough not impressive, the improvement brought by constraints is substantial, given the fact that there are only 37 constraints which involve 97 shots (compared with the total of 376 shots to be named), and many constraints are on anchors, the accuracy on which is already very high. Between the two types of aggregates used to implement constraints, average-aggregate is slightly better than maximum-aggregate. If we look at top-2 or top-3 names, however, **LM+MaxCons** and **LM+AvgCons** have about the same performance as **LM**. This implies that constraints can help boost the most likely candidate name into high ranks, but are not as effective in penalizing the unlikely names into low ranks.

Another interesting experiment investigates the contributions of different types of constraints and the strategies of combining them. Specifically, we study the performance of using the constraints from image similarity (**LM+ImgCons**), from speaker ID (**LM+SidCons**), their union (**LM+UniCons**), and their intersection (**LM+IntCons**). We have also implemented a method using "oracle constraints", which are all the constraints that can be derived from the truth. For example, if the truth shows that two shots



**Figure 7: Performance of using different constraints**

have the same person, we add an oracle constraint between the two shots. This method (**LM+OraCons**) gives the *upper bound* on what can be achieved using constraints. The performance at top-1 name of using different classes of constraints implemented by average-aggregate is shown in Figure 6.

As we can see, using constraints from a single source of information achieves only a trivial improvement over **LM**, and the improvement is not consistent across different categories of shots. A likely explanation is that constraints derived from a single source are neither accurate nor sufficient. This is further confirmed by the observation that using the union of these two types of constraints still does not improve the performance. In this case, although there are more constraints, there are also more errors in them, which eliminate the contribution of the correct constraints. In comparison, using the intersection of the two types of constraints improves the performance the most, since the shared constraints are of high accuracy. This suggests that the accuracy of the constraints is more important than the completeness. It is surprising to learn that using oracle constraints do not perform significantly better than using the intersection of constraints. This echoes the fact that even correct constraints do not always improve the predictions of **LM** but may degrade it, especially when the predictions of **LM** are of poor quality. Thus, using more constraints does not guarantee better performance.

## 7. CONCLUSION

We have described a machine learning model for naming every individual person in broadcast news videos. Trained based on some manually named people, this model predicts the most likely name of each individual who is giving a monologue speech based on features extracted from multiple modalities including speech, overlaid text, transcript, etc. Equivalence constraints relating the names of different persons are used to refine the predictions. Experiments have shown that this approach significantly outperforms a non-learning alternative in terms of the accuracy of the names assigned, particularly when the constraints are applied.

As we see in Section 6.1, nearly one forth (122 out of 498) of the monologue shots contain anonymous people or people with name only existing in overlaid text, which our approach currently cannot hope with. It is desirable in our future work to improve our approach to determine whether the speaker in a monologue shot is anonymous, and if not, whether his/her name appears in closed-caption or in overlaid text. We expect a deeper analysis of speaker identities and the context in transcript where names are mentioned

will be crucial for this problem. A more ambitious goal is to extend our work to address non-monologue shots, such as shots containing a person who is not speaking, or shots where multiple speaking people appear simultaneously. Since speaker identification is not applicable or even misleading in these cases, the use of face recognition or facial similarity is inevitable. An alternative of the facial similarity is the image similarity of region bounding the faces, which has been proven effective in [15].

At algorithmic level, one extension of our current work is to replace SVM with other learning algorithms, such as logistic regression and decision tree. Moreover, since the contribution of constraints are not significant in our experiment, another future work is to integrate the constraints with the classifier in a more justified manner, where a promising approach is the discriminative learning framework with pair-wise constraints in [7].

## 8. ACKNOWLEDGEMENTS

This research is partially supported by the Advanced Research and Development Activity (ARDA) under contract number H98230-04-C-0406 and NBCHC040037.

## 9. REFERENCES

- [1] Bikel, D. M., Miller, S., Schwartz, R., and Weischedel, R., Nymble: a high-performance learning name-finder. In *Proc. 5th Conf. on Applied Natural Language Processing*, 1997, pp. 194-201.
- [2] Berg, T., Berg, A., Edwards, J., Maire, M., White, R., Teh, Y.W., Miller, E., Forsyth, D. Names and Faces in the News. In *Proc. of Computer Vision and Pattern Recognition*, Vol.2, pp. 848-854, 2004.
- [3] Burges, C. J. C. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121-167, 1998.
- [4] Gauvain, J.L., Lamel, L., and Adda, G. The LIMSI broadcast news transcription system. *Speech Communication*, 37(1-2): 89-108, 2002.
- [5] Hauptmann, A., Witbrock, M. Informedia: News-on-Demand Multimedia Information Acquisition and Retrieval. *Intelligent Multimedia Information Retrieval*, Mark T. Maybury, Ed., AAAI Press, pp. 213-239, 1997.
- [6] Houghton, R. Named Faces: Putting Names to Faces. In *IEEE Intelligent Systems Magazine*, 14(5): 45-50, 1999.
- [7] Rong, Y., Zhang, J., Yang, J. and Hauptmann, A. A Discriminative Learning Framework with Pair-wise Constraints for Video Object Classification. In *Proc. of Computer Vision and Pattern Recognition*, Vol.2, pp. 284-291, 2004.
- [8] Sato, T., Kanade, T., Hughes, E. K., Smith, M. A., Satoh, S. Video OCR: Indexing digital news libraries by recognition of superimposed caption. *ACM Multimedia Systems*, 7(5): 385-395, 1999.
- [9] Satoh, S., Y., Kanade, T. NAME-IT: Association of Faces and Names in Video. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, 1997, pp. 368-373.
- [10] Schneiderman, H., Kanade, T. Object detection using the statistics of parts. *Int'l J. of Comp. Vision*, 56(3): 151-177, 2002.
- [11] Snoek, C.G.M. and Hauptmann, A. *Learning to identify TV news monologues by style and context*. Technical Report, CMU-CS-03-193, Carnegie Mellon University, 2003.
- [12] TRECVID: TREC Video Retrieval Evaluation: <http://www-nlpir.nist.gov/projects/trecvid/>.
- [13] Yang, J., Chen, M.Y., Hauptmann, A. Finding Person X: Correlating Names with Visual Appearances. *Int'l Conf. on Image and Video Retrieval*, 2004. (To appear)
- [14] Zhang, H.J., Tan, S.Y., Smoliar, S.W., Gong, Y.H. Automatic parsing and indexing of news video. In *Multimedia Systems*, 2(6): 256-266, 1995.
- [15] Zhang, L., Chen, L.B., Li, M.J., Zhang, H.J. Automated annotation of human faces in family albums. In *Proc. of 11th ACM Int'l Conf. on Multimedia*, 2003, pp. 355 - 358.
- [16] Zhao, W.Y., Chellappa, R., Phillips, P.J., and Rosenfeld, A. Face Recognition: A Literature Survey. In *ACM Computing Survey*, Dec. Issue, pp. 399-458, 2003

## APPENDIX I:

Multi-modal features for shot-name pair  $\langle s, n \rangle$

Modality	Feature	Description
Speaker ID of shot $s$ ( $sid$ )	<i>speaker_cross_story</i>	whether $sid$ appears in multiple stories
	<i>speaker_dominant</i>	whether $sid$ is the speaker ID with maximal coverage in the story
	<i>speaker_utter_name</i>	whether the speech of $sid$ utters name $n$ in the story
	<i>speaker_span</i>	number of adjacent shots where $sid$ is the primary speaker ID
Video OCR	<i>vocr_len</i>	number of characters in VOCR text, zero if VOCR does not exist
	<i>vocr_edit_dist</i>	edit distance between the VOCR text and $n$
Temporal structure	<i>str_story_len</i>	story length in terms of the number of shots in it
	<i>str_offset</i>	order of $s$ from the start or end of the shot sequence of the news story
Name position	<i>name_position</i>	whether $n$ appears before, after, or within the boundaries of $s$
	<i>name_order</i>	order of $n$ among all the candidate names appearing before (or after) $s$
Transcript clues	<i>trans_anchor_shot</i>	whether the transcript clues suggest that $s$ is an anchor shot
	<i>trans_reporter_shot</i>	whether the transcript clues suggest that $s$ is a reporter shot
	<i>trans_anchor_name</i>	whether the transcript clues suggest that $n$ is an anchor's name
	<i>trans_reporter_name</i>	whether the transcript clues suggest that $n$ is a reporter's name