On Clustering Users' Behaviors in Video Sessions

Sylvain Mongy
LIFL-UMR USTL-CNRS 8022
Laboratoire d'Informatique
Fondamentale de Lille
Université de Lille 1, France
mongy@lifl.fr

Chabane Djeraba
LIFL-UMR USTL-CNRS 8022
Laboratoire d'Informatique
Fondamentale de Lille
Université de Lille 1, France
djeraba@lifl.fr

Dan Simovici
University of Massachusetts Boston
Department of Computer Science
Boston, Massachusetts 02125, USA,
dsim@cs.umb.edu

Abstract

We study the extraction of characteristics of user behavior in video session encoded as stochastic matrices of finite Markov chain. These behaviors are clustered using a dissimilarity based on the Kullbach-Leibler divergence between probability distributions. The center of each cluster is regarded as the model that generates the behaviors assigned to the cluster. This choice is based on the relationship that we establish between the dissimilarity between the behavior and the model, and the probability that the model generates the behavior. Experimental results that evaluate the quality of the clustering validate our choice of the models.

1 Introduction

Video support has become an important information carrier in both raw and commercial data with the advent of significant progress in information transfer. Increasingly, video data is the main vehicle for information publishing. The generalization of video data is strongly linked to web development and to audiovisual production techniques. These developments raise many challenges and are relatively unexplored.

Other types of multimedia data such as images or text are persistent and are used for satisfying the end users and retain their attention. For example, the architecture of web sites is such that visitors find the sought information as quickly as possible; advertising items are placed such that they have sufficient visibility but do not interfere with the information seeking activity.

The increasing importance of video data generates new type of user behavior. This type of data, especially complex, requires new analysing tools that facilitate the understanding of their usage modalities and are able to suggest ways of making them more attractive for the users. One notable area of application is the production of advertising videos which requires new tools for behavior analysis.

We propose here a new technique for extracting the characteristics of user behavior in video sessions. Starting from the action logs of the sessions (play, fast forward, rewind, etc.) we construct viewings that correspond to the sequences of user actions and to their durations during a viewing session. After that, we cluster these viewings in order to extract types of observed behaviors. An analysis by a domain specialist allows the assignment of a succinct description such as a "fast viewing of the video", "viewing of a specific video sequence", or "a complete viewing of the video". These behaviors should allow professionals to evaluate the impact of videos on consumers.

The technique that we use is based on the representation of behaviors as first-order Markov chains [1]. These models are defined by considering finite graphs having user actions as vertices; edges of these graphs represent transitions between these actions. They are labelled by probabilities of these transitions extracted from real user behaviors. The usefulness of finite Markov models for this study has been recognized in the literature for a rather long time. Pioneering work in this direction can be traced back to [4] where simple, two-state Markov chains help in formulating the idea of effective fast-forward/rewind service, a video server architecture that accommodates with a high probability within a limited bandwidth a large number of users that need to have video access on demand. In [5] similar Markov models are used in the development of an algorithm that integrates scalable compression techniques with placement algorithms for disk arrays in order to provide service support for fast-forward and rewind operations in video servers. The increasing importance of interactive video-on-demand that requires VCR-like functions was studied in [6] using Markov chains in order to develop an evaluation tool for a video system design. All these investigations are focused on system architectures and on quality-of-service issues. Our focus is on the classification of users' behaviors.

We cluster users' behaviors by applying the k-means algorithm [3]; the dissimilarity involved in the algorithm is a symmetrization of the well-known Kullbach-Leibler dissimilarity, suitably modified in order to deal with null components of certain probability distributions. This dissimilarity is important for our study since, as we show in Section 3, the probability that a model generates a certain behavior is large, when the Kullbach-Leibler dissimilarity between the distribution of the model and the empirical probability distribution (of user behavior) is small.

Our preliminary tests on real data yield good results. The clustering technique used in the paper produces viewing models that are quite distinct and cover the diversity of observed behaviors.

The paper is structured as follows. After a presentation of the current state of research in Section 2, we introduce in Section 3 theoretical concepts related to our models and their link to the Kullbach-Leibler dissimilarity. Then, in Section 4 we present the results of experiments involving our classification technique of user behavior using k-means clustering. Our conclusions are presented in Section 5.

	PLAY	PAUSE	JUMP	FFW	RWD	STOP
PLAY	0,9	0,08	0	0	0	0,02
PAUSE	0,21	0,54	0,07	0,08	0,05	0,05
JUMP	0	1	0	0	0	0
FFW	0	0,88	0	0,12	0	0
RWD	0	0,85	0	0	0,15	0
STOP	0	0	0	0	0	1

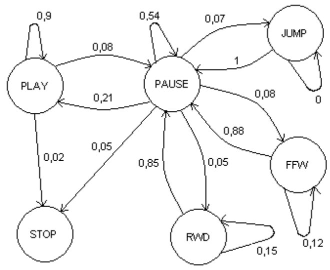

Figure 1. Transition Matrix and Its Transition Graph

In our case the sequences are relatively simple. Indeed, there are only six types of actions: play, pause, stop, fast forward, fast rewind, jump. This limited number of possibilities allows us to represent viewings as Markov chains with small numbers of states [1]. This approach allows us to represent viewings in a compact manner. To preserve the information carried by the sequences of actions, we need to account for the time spent by the user in each of these states. This is accomplished by discretizing the time and by assuming that a transition is performed each second. If a user spends 10 seconds playing a sequence, we record this as 10 transitions play $\rightarrow$ play. This idea introduced in [7] allows us to take the time in consideration without adding an extra parameter to the model or by splitting the states. Figure 1 presents the transition matrix of a viewing and its corresponding transition graph.

It is interesting to observe the simplicity of this representation that reduces a viewing to a 6×6 stochastic matrix. We discuss next the usage of the Kullbach-Leibler distance in the study of these data.

2 Models of Users' Behaviors for Video Data

Viewings are represented by sequences of actions of the users while watching a video sequence: play, pause, etc., together with their durations. For example, an user may have watched the video for 10 seconds before pausing for 5 seconds, then fast forwarding for 30 seconds and concluding the viewing by watching 20 seconds more. This corresponds to the sequence

((PLAY, 10), (PAUSE, 5), (FAST_FORWARD, 30), (PLAY, 20)).

Treating viewing under this raw form is not the best approach. Indeed, the comparisons between sequential data are difficult and expensive regardless of the technique which is applied (alignment of sequences [10, 11, 12], searching for the longest common subsequence [13], or other methods).

3 Models and The Kullbach-Leibler Dissimilarity

A *model* is an $n \times n$-stochastic matrix $P = (p_{ij})$, where $p_{ij} \geq 0$ for $1 \leq i, j \leq n$ and $\sum_{j=1}^{n} p_{ij} = 1$ for every i, $1 \leq i \leq n$. An element p_{ij} of this matrix is interpreted as the probability of a transition from a state s_i to a state s_j.

Let $\mathbf{s} = (s_{i_1}, \ldots, s_{i_\ell})$ be a sequence of states produced by an experiment. For example, such an experiment could be the succession of commands issued by an user who is watching a video:

$$(\text{play}, \text{stop}, \text{rewind}, \text{play}, \text{stop}, \ldots).$$

Suppose that there are n possible states of the experiment and that the number of transitions from state s_i to state s_j observed in this sequence is c_{ij}. The frequency matrix of the sequence $\mathbf{s}$ is the matrix $C(\mathbf{s}) = (c_{ij})$. Note that $C(\mathbf{ss}') = C(\mathbf{s}) + C(\mathbf{s}')$.

The *probability* that a sequence $\mathbf{s}$ is produced by a model P is the number $p(\mathbf{s}|P) = \prod_{i=1}^{n} \prod_{j=1}^{n} p_{ij}^{c_{ij}}$. This implies:

$$\log p(\mathbf{s}|P) = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \log p_{ij}.$$

Let $n_i = \sum_{j=1}^{n} c_{ij}$ be the frequency of the state s_i in the sequence $\mathbf{s}$. Note that the matrix $F(\mathbf{s}) = \left(\frac{c_{ij}}{n_i}\right)$ is an $n \times n$ stochastic matrix. This allows us to write

$$\log p(\mathbf{s}|P) = \sum_{i=1}^{n} n_i \sum_{j=1}^{n} f_{ij} \log p_{ij},$$

where $f_{ij} = \frac{c_{ij}}{n_i}$ for $1 \leq i, j \leq n$.

We need to evaluate how different the stochastic matrices $F(\mathbf{s})$ and P are and this can be achieved using a dissimilarity (or a distance, whenever possible) between these matrices.

Let $\mathbf{u} = (u_i), \mathbf{v} = (v_i)$ be two n-dimensional stochastic vectors, that is two-vectors with non-negative components such that $\sum_{i=1}^{n} u_i = \sum_{i=1}^{n} v_i = 1$. The Kullbach-Leibler dissimilarity between $\mathbf{u}$ and $\mathbf{v}$ is

$$d_{KL}(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^{n} u_i \log \frac{u_i}{v_i}.$$

It is easy to verify that $d_{KL}(\mathbf{u}, \mathbf{v}) \geq 0$ and that $d_{KL}(\mathbf{u}, \mathbf{v}) = 0$ if and only if $\mathbf{u} = \mathbf{v}$.

We begin by showing a linkage between a matrix dissimilarity generated by the Kullbach-Leibler dissimilarity between probability distributions and the probability that a sequence is generated by a model.

Let $F(\mathbf{s})$ be the frequency matrix of a sequence $\mathbf{s}$ and let P be a stochastic matrix. Denote by $\mathbf{f}_i$ and $\mathbf{p}_i$ the rows of these matrices (which are probability distributions).

The dissimilarity $D_{KL}(F(\mathbf{s}), P)$ between the matrices $F(\mathbf{s})$ and P is defined as:

$$D_{KL}(F(\mathbf{s}), P) = \sum_{i=1}^{n} n_i d_{KL}(\mathbf{f}_i, \mathbf{p}_i).$$

Theorem 3.1 *The quantity $D_{KL}(F(\mathbf{s}), P) + \log p(\mathbf{s}|P)$ is constant for all models P.*

Proof. Starting from the definition of $D_{KL}(F(\mathbf{s}), P)$ we can write:

$$\begin{aligned}
&D_{KL}(F(\mathbf{s}), P) \\
&= \sum_{i=1}^{n} n_i d_{KL}(\mathbf{f}_i, \mathbf{p}_i) \\
&= \sum_{i=1}^{n} n_i \sum_{j=1}^{n} f_{ij} \log \frac{f_{ij}}{p_{ij}} \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} n_i f_{ij} \log f_{ij} - \sum_{i=1}^{n} \sum_{j=1}^{n} n_i f_{ij} \log p_{ij} \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} n_i f_{ij} \log f_{ij} - \log p(\mathbf{s}|P) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \log \frac{c_{ij}}{n_i} - \log p(\mathbf{s}|P),
\end{aligned}$$

which justifies our statement. ∎

Corollary 3.2 *The likelihood that a model P generates a sequence $\mathbf{s}$ is decreasing when the dissimilarity between the matrix of the model and the frequency matrix of a sequence $\mathbf{s}$ is increasing.*

Proof. This statement follows immediately from Theorem 3.1. ∎

The Kullbach-Leibler dissimilarity is inconvenient when there are zero entries in one of the matrices $F(\mathbf{s})$ or P. In this case we use a Laplace-like approximation of this dissimilarity. Each zero entry is replaced by a small number ϵ. To maintain the stochastic character of the matrices we need to multiply each non-zero element by a corresponding quantity. Suppose, for example, that $\mathbf{v} = (v_1, \ldots, v_n)$ is a stochastic vector that has k zero entries. Then, replacing these entries by ϵ means that we need to multiply each of the remaining $n - k$ non-zero entries of this vector by $\alpha = 1 - k\epsilon$. For example, if the zero entries of $\mathbf{v}$ occupy the last k positions we shall replace $\mathbf{v}$ by $\mathbf{v}' = (\alpha v_1, \ldots, \alpha v_{n-k}, \epsilon, \ldots, \epsilon)$. Of course, we need to choose $\epsilon < \frac{1}{k}$.

To apply this treatment to an entire matrix it suffices to take $\epsilon < \frac{1}{k_{\max}}$, where k_{max} is the largest number of zero entries in a line of the matrix. Then, each line i of the matrix must be multiplied by $1 - k_i\epsilon$, where k_i is the number of zero entries in the line i and the zeros of this line replaced by ϵ. We adopted the value $\epsilon = 0.01 < \frac{1}{36}$.

4 Experimental Results

Our objective is to group together similar viewings using a k-means algorithm based on dissimilarity derived from the Kullbach-Leibler dissimilarity. If $F(\mathbf{s}), F(\mathbf{s}')$ are two frequency matrices, we define the dissimilarity δ_{KL} by

$$\delta_{KL}(F(\mathbf{s}), F(\mathbf{s}')) = d_{KL}(F(\mathbf{s}), F(\mathbf{s}')) + d_{KL}(F(\mathbf{s}'), F(\mathbf{s})).$$

Selecting an optimality criterion and finding the optimum number of clusters is a special challenge of the k-means algorithm. We adopt the approach proposed by [2] which defines the quality of a clustering as the ratio between the inter-dissimilarity and the intra-dissimilarity between clusters, as we explain in detail in Section 4.2. The centroids of the resulting clusters play the role of the models for the sequences that belong to their respective clusters, a choice that is based on Corollary 3.2.

4.1 Data Collection

All actions of the users are traced and written on log files. To store these data, we define a set of specialized XML tags. Pre-processing of data allows us to construct viewings by grouping chronologically the actions of the users and the videos viewed. Then, we proceed with the clustering.

To obtain relevant data we developed a search engine allowing viewing of advertising clips for movies. This tool offers basic searches (based on director, actors, date of release, etc.) and allows the user to view the clips that the search returns. Ten viewers were invited to perform searches using this tool and a questionnaire was attached in order to guide them and thus collect diverse and interesting data. The role of the questionnaire was to propose different situations to the users and to stimulate them to view the clips as completely as possible. Typical questions included were: "from which movie was this image extracted?", "which is the best advertising?", "which are the action movies that occur on this list?", etc.

The data we collected are quite close to a sample of real production data. Next, we discuss the results obtained by clustering.

4.2 Optimal Clusterings

To evaluate the quality of the clustering constructed by the application of the k-means algorithm we compute two numbers: the intra-cluster dissimilarity and the inter-cluster dissimilarity. The intra-cluster dissimilarity, denoted by *IntraDiss* measures how tightly the clusters are grouped around their centers by evaluating the average dissimilarity between the center of the cluster and the members of the cluster. If k is the number of clusters, c_i is the center of the i-th cluster C_i, then the intra-cluster dissimilarity is given by:

$$IntraDiss = \frac{\sum_{j=1}^{k} \frac{\sum_{x_i \in C_j} d_{KL}(x_i, c_j)}{|C_j|}}{k} \quad (1)$$

The inter-cluster dissimilarity evaluates the separation between different clusters. For each group, one evaluates the average dissimilarity between its center and the objects that belong to every other cluster. The average is taken over the set of clusters and yields the inter-cluster dissimilarity:

$$InterDiss = \frac{\sum_{j=1}^{k} \frac{\sum_{x_i \notin C_j} d_{KL}(x_i, c_j)}{|U| - |C_j|}}{k}, \quad (2)$$

where U is the entire collection of objects.

To optimize the clustering it is desirable to have have compact and well-separated clusters which imply large values of the inter-cluster dissimilarity and small values for the intra-cluster dissimilarity, which suggest that local maxima for the Ray-Turi index:

$$r = \frac{InterDiss}{IntraDiss} \quad (3)$$

are desirable (see [2]). The table 1 presents the values obtained for these quantities (*InterDiss*, *IntraDiss* and r) for different values of k, as well as the ratios of these values averaged over 40 runs for each of the values of k between 3 and 10.

k	InterDiss	IntraDiss	r
3	42.60	15.66	2.72
4	29.80	12.71	2.34
5	22.83	9.17	2.49
6	18.09	8.23	2.20
7	15.60	6.86	2.28
8	14.49	5.86	2.47
9	13.34	4.63	2.88
10	12.17	4.34	2.80

Table 1. Cluster quality

The figure 2 shows the dependency of r on the number of clusters k that serves as input for the k-means algorithm. One observes a local maximum of r for $k = 5$. This corresponds to one of our previous analysis of the viewings [8, 9].

We observe that the value of r is a local maximum of the curve for $k = 5$. This maximum corresponds to the optimal value of the number of classes. The interest of this result is the confirmation of our previous experiments in which we found 4 to 5 types of behavior during video watching [8]. The Figure 3 shows these five models obtained as cluster centers.

The first and the fourth models correspond to a fast closing of the video session after a few seconds of viewing. The second model has an average value for the probability of the transition play $\rightarrow$ play and a relatively important value for the transitions play $\rightarrow$ jump and jump $\rightarrow$ play corresponding to a fragmentary watching of the video.

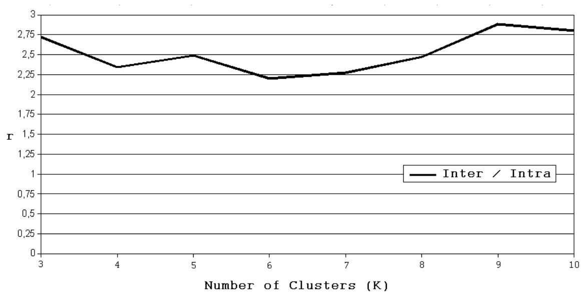

Figure 2. Cluster quality as a function of the number of clusters

The third model is similar to the second, with lower probabilities of returning to the *play* state, which corresponds to a fast perusing of the video.

Finally, the last model has a rather high probability of the transition play $\rightarrow$ play, corresponding to a detailed watching of the video.

5 Conclusion et Future Work

We present an analysing technique for users' behavior during watching of video sequences. This technique is based on a succinct representation of these behaviors using Markov models which allows the preservation of the most important information related to video behaviors and facilitates their comparative study.

Both viewing records and models are represented by stochastic matrices and we show that the probability that a behavior is generated by a model varies inversely with a certain dissimilarity between the model and the behavior that is defined starting with the Kullbach-Leibler dissimilarity between probability distributions. Thus, by clustering the models, using a k-means algorithm we adopt the centers of the clusters as models for user behaviors. The quality of the clusterings is assessed using the Ray-Turi criterion. The behavior types that we identified elsewhere [8] turns out to be a local maximum of the Ray-Turi index, thus conforming the validity of our approach.

We intend to examine user behavior using other technique that involve spectral properties of the stochastic matrices involved and the asymptotic behaviors that can be attached to these Markov chains.

	PLAY	PAUSE	JUMP	FWD	RWD	STOP
PLAY	7%	34%	6%	0%	0%	50%
PAUSE	4%	0%	15%	0%	0%	80%
JUMP	48%	16%	23%	0%	0%	10%
FWD	0%	0%	0%	100%	0%	0%
RWD	0%	0%	57%	0%	42%	0%
STOP	100%	0%	0%	0%	0%	0%

	PLAY	PAUSE	JUMP	FWD	RWD	STOP
PLAY	50%	23%	23%	0%	0%	2%
PAUSE	6%	0%	6%	0%	3%	84%
JUMP	81%	0%	2%	0%	0%	15%
FWD	100%	0%	0%	0%	0%	0%
RWD	0%	0%	3%	0%	1%	95%
STOP	100%	0%	0%	0%	0%	0%

	PLAY	PAUSE	JUMP	FWD	RWD	STOP
PLAY	38%	6%	40%	0%	0%	14%
PAUSE	99%	0%	0%	0%	0%	0%
JUMP	42%	13%	20%	0%	0%	23%
FWD	100%	0%	0%	0%	0%	0%
RWD	0%	0%	0%	0%	100%	0%
STOP	0%	0%	0%	0%	0%	100%

	PLAY	PAUSE	JUMP	FWD	RWD	STOP
PLAY	1%	98%	0%	0%	0%	0%
PAUSE	18%	0%	11%	0%	1%	68%
JUMP	0%	0%	60%	0%	0%	39%
FWD	0%	0%	0%	0%	0%	100%
RWD	0%	0%	0%	100%	0%	0%
STOP	0%	0%	0%	0%	0%	100%

	PLAY	PAUSE	JUMP	FWD	RWD	STOP
PLAY	74%	2%	23%	0%	0%	0%
PAUSE	50%	0%	0%	0%	50%	0%
JUMP	22%	0%	0%	0%	22%	55%
FWD	0%	0%	0%	100%	0%	0%
RWD	0%	0%	0%	0%	15%	84%
STOP	100%	0%	0%	0%	0%	0%

Figure 3. Results for $k = 5$

References

[1] P. Brémaud: *Markov Chains*, Springer Verlag, 2001.

[2] S. Ray and R. H. Turi: *Determination of number of clusters in K-means clustering and application in colour image segmentation*, invited paper, in N. R. Pal, A. K. De and J. Das (eds), Proceedings of the 4th International Conference on Advances in Pattern Recognition and Digital Techniques (ICAPRDT'99), Calcutta, India, 27-29 December, 1999, Narosa Publishing House, New Delhi, India, pp 137-143.

[3] J. MacQueen: *Some methods for classification and analysis of multivariate observations*, Proc. Fifth Berkeley Symp. University of California Press 1, 1966, pp 281-297.

[4] Jayanta Day-Sircar and James D. Salehi and James K. Kurose and Don Towsley: *Providing VCR Capabilities in Large-Scale Video Servers*, Proceedings of the ACM Mulimedia 94, 1994, San-Francisco, CA, pp. 25-32.

[5] Prashant J. Shenoy and Harrick M. Vin, *Efficient Support for Scan Operations in Video Servers*, Proceedings

of the ACM Mulimedia 95, 1995, San Francisco, CA, pp. 131-140.

[6] Victor O. K. Li and Wanjiun Liao and Xiaoxin Qiu and Eric W. M. Wong: *Performance Model of Interactive Vido-on-Demand Systems*, IEEE Journal on Selected Areas in Communications, vol. 14, no. 6, August 1996, pp. 1099-1109.

[7] Philip Branch and Greg Egan and Bruce Tonkin: *Modeling interactive behaviour of a video based multimedia system*, IEEE International Conference on Communications, June, 1999, Vol.2, pp. 978-982.

[8] Sylvain Mongy: *Analyse du Comportement des utilisateurs exploitant une base de vidéos*, Proceedings of the 6th 'Journées Francophones Extraction et Gestion de Connaissances' (EGC 2006), RNTI-E-6 Cépaduès-Editions, 2006, pp. 391-402.

[9] Sylvain Mongy and Chabane Djeraba: *A Study on video viewing behavior. Application to movie trailer miner*, Proceedings of the IS&T/SPIE 19th Annual Symposium Electronic Imaging Science and Technology (SPIE 2007). 28 January - 01 February 2007, San Jose California USA (in print).

[10] Weinan Wang and Osmar R. Zaïane: *Clustering Web Sessions by Sequence Alignment*, Proceedings of the 13th International Workshop on Database and Expert Systems Applications, DEXA 2002, pp. 394 - 398.

[11] Birgit Hay and Geert Wets and Koen Vanhoof: *Clustering navigation patterns on a website using a Sequence Alignment Method*, Intelligent techniques for web personalization: IJCAI 2001 17th International Joint Conference on Artificial Intelligence, August 4, Seattle, Wash., USA, s.l., , 2001, pp. 1-6.

[12] Kevin Charter and Jonathan Schaeffer and Duane Szafron: *Sequence Alignment Using FastLSA*, International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences (METMBS'2000), 2000, pp. 239-245.

[13] Arindam Banerjee and Joydeep Ghosh: *Clickstream Clustering using Weighted Longest Common Subsequences*, Proceedings of the Web Mining Workshop at the 1st SIAM Conference on Data Mining, Chicago, April 2001, pp. 33-40.