HIERARCHICAL UNSUPERVISED DISCOVERY OF USER CONTEXT FROM MULTIVARIATE SENSORY DATA

Okko Räsänen

Department of Signal Processing and Acoustics, Aalto University, Finland

ABSTRACT

A system capable for purely unsupervised learning of sensory context models is presented in this work. The system is based on discovery of short-term activity motifs from the sensory data and statistical analysis of these motifs on a larger time scale. Detected context segments are then clustered into high-level context categories and the data corresponding to these categories are used to train on-line classifiers for different contexts. Experiments show that the method is capable of segmenting sensory recordings into epochs of high-level environmental contexts based purely on audio signal, and that the classifiers trained from the obtained segments are selective towards specific contexts.

Index Terms— context recognition, machine learning, unsupervised learning

1. INTRODUCTION

Technological development has reached a point where compact mobile devices are able to sense their environment using a variety of built-in sensors, providing massive amounts of data from the surrounding world. However, the raw data as such is meaningless. Therefore one of the aims in context-aware computing is to infer higher-level abstract representations of the surrounding context from the sensory data that would provide useful information regarding the current use situation of the device (e.g., location such as *shop* or *home* or activity such as *walking*). Majority of the previous work in user context recognition has used supervised methods to train separate classifiers for different physical activities and auditory contexts of interest. For example, Pärkka et al. [1] and Ermes et al. [2] have studied classification of physical activity from accelerometer data using pre-trained classifiers. Auditory context recognition with pre-trained audio classifiers has also been studied (e.g., [3]). The general finding of the studies is that the context recognition performance achieves relatively good levels when the training data has close correspondence to the actual testing conditions. When controlled in-lab data sets are evaluated in unconstrained situations, performance drops significantly [2,3].

In order to overcome the limitations of pre-trained classifiers and to allow user-specific adaptation of the models, unsupervised methodology for context discovery can be utilized. For example, discovery of high-level contexts has been studied in the work of Clarkson & Pentland [4], who present a hierarchical HMM framework where low-level HMMs represent temporally brief events such as door closings and cash register beeps, whereas high-level HMMs are used to model sequences of low-level events ("high-level contexts"). Also, Krause et al. [5] have introduced an adaptive mobile phone system that learns user contexts automatically from numerous sensory streams and adapts the system to the user behavior in these contexts.

In this work, a novel approach for unsupervised learning of high-level user contexts from any generic sensory data is presented. The system combines unsupervised discovery of short-term sensory motifs to unsupervised acquisition of high-level context models in a hierarchical framework that is computationally feasible for platforms with low computational resources. The basic idea is to first discover statistically significant recurring structures in sensory streams and then to analyze the presence of these structures at a larger time-scale in order to find internally coherent segments of sensory activity. These segments are then clustered into context categories and on-line recognizers are trained for the categories using the discovered segments as the training data. Performance of the system is demonstrated using realistic audio data recordings from various everyday activities and locations.

2. DATA

Palantir Context Data Library 2003 [1,2] used in the experiments consist of multisensory recordings (a total of 18 sensors) from 16 test subjects. Recordings were made with a portable recording system while the test subject was performing a variety of everyday activities. Each recording is 2-3 hours long and all subjects follow a pre-defined scenario that contains both indoor and outdoor activities at different locations. The following activities are annotated in the data: throwing a ball, biking, playing croquet and football, lying, Nordic walking, rowing, running, sitting, standing, swinging, and walking. In addition, location information is available in terms of following categories: office, bus, indoors, lake, library, park, restaurant, shop, street, and unknown. The annotation was made during recordings by a separate observer.

Since experiments with multisensory learning are out of the scope of this paper, we will use only audio signals (f_s = 22050 Hz) recorded from a microphone attached to the strap of a backpack in order to demonstrate the feasibility of the basic system.

3. METHODS

The unsupervised context learning process consists of five main stages: 1) pre-processing of the sensory data into vector-quantized (VQ) discrete sequences, 2) learning of motifs (recurring patterns) from the sequential data, 3) segmentation of the data into temporal epochs by studying the prevalence of discovered motifs and VQ-elements in the data, 4) clustering of the detected segments into context categories, and 5) training of on-line classifiers for low-level features of each high-level context category. Fig. 1 shows a schematic view of the entire process. Although audio is used in this work, the framework generalizes to any sensory data (e.g., acceleration) with sensor specific feature extraction front-end.

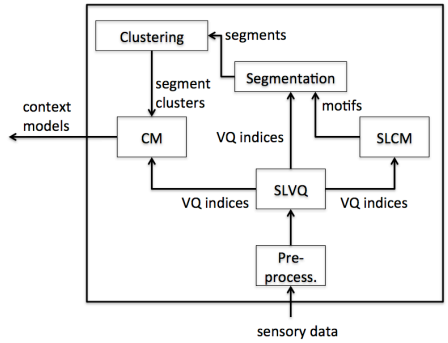

Fig. 1: A schematic view of the unsupervised context learning system. Long-term statistical analysis of pattern motifs and VQ indices is used to discover high-level context classes, for which on-line classifiers can be then trained.

3.1. Pre-processing

Audio is pre-processed by resampling the signals to 16 kHz sample rate and then extracting standard 36 MFCC features (12 static, delta, and deltadelta) with a Hamming window of 32 ms and a step size of 10 ms. Energy is not used since it is not a reliable feature in real world recordings using a mobile microphone.

Then a VQ codebook is created for the audio using self-learning vector quantization (SLVQ [6]) algorithm that adjusts the number of clusters according to the data properties. Then all MFCCs are vector quantized using the codebook. After pre-processing, the sensory data is represented by a discrete sequence $X = [a_1, a_2, \ldots, a_N]$ of length N, where each a_i belongs to the alphabet $\mathbf{A} = \{1, 2, \ldots, N_A\}$.

3.2. Unsupervised learning of motifs

The unsupervised learning of motifs from the data is performed using the Self-learning Concept Matrices (SLCM) algorithm that was originally presented in the context of unsupervised learning of word forms from continuous speech [7].

When a sequence $X = [a_1, a_2, \ldots, a_n]$ is used as input, the subsequence Ω of the first L elements $\Omega(1) = \{a_1, a_2, \ldots, a_L\}$ of the sequence is chosen and the transition frequencies between element pairs $f(a_i,a_j)$, $a \in [1, 2, \ldots, N_A]$, at lags $k_d \in \boldsymbol{k}$ in $\Omega(1)$ are stored into transition frequency matrices $f_c(a_i \mid a_j , k_d)$, where $c = 1$ for the first model, i.e., a separate matrix is created for each lag. Then the frequency matrices are normalized into transition probability (TP) matrices P^s:

$$P_c^{\mathrm{S}}(a_j \mid a_i, k_d) = f_c(a_j \mid a_i, k_d) / \sum_{j=1}^{N_A} f_c(a_j \mid a_i, k_d) \qquad (1)$$

Then the analysis window is shifted S elements forward to position $\Omega(\mathrm{T} = 2) = \{a_{1+T*S}, a_{2+T*S}, \ldots, a_{L+T*S}\}$ and the previously learned models c are used to compute the transition probabilities of the new sequence by using the learned models:

$$A(c,n) = \frac{1}{K} \sum_{d=1}^{K} P_c^{\mathrm{S}}(\Omega[n] \mid \Omega[n-k_d], k_d) \qquad (2)$$

i.e., the mean of TPs is computed across all lags $\boldsymbol{k}$. Finally, the mean probability of each model in $\Omega(T)$ is computed:

$$\hat{A}(c,T) = \frac{1}{L} \sum_{n=1}^{L} A(c,n) \qquad (3)$$

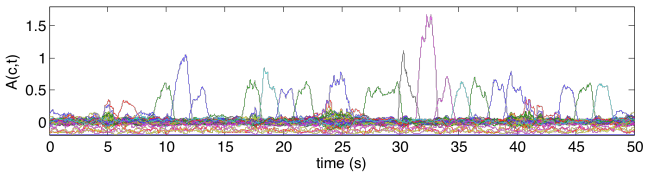

Fig. 2: An example of motif detector output for the audio stream. Activations of different models are shown in different colors.

Now, if the activation $\hat{A}(c,T)$ of any single model exceeds a pre-defined threshold δ, the TPs of the most activated model are updated according to (1) using the transitions in the sequence $\Omega(T)$. If no sufficiently high activation is achieved, a new model c_m is created using the transitions in $\Omega(T)$. The window is then again shifted L elements and the new subsequence $\Omega(T+1)$ is recognized using the learned models. This windowing process is repeated for the duration of entire training signal, leading to the learning of a number of models for patterns in the input sequence.

After the models have been learned, their activity during the signal is re-estimated. In order to enhance contrast between the learned models, the probability that a specific transition from a_i to a_j occurs in the case of model c and lag k, instead of any other models, is incorporated into the *activation matrix P* by having:

$$P_c(a_j \mid a_i, k_d) = P_c^{\mathrm{S}}(a_j \mid a_i, k_d) / \sum_{g=1}^{N_C} P_g^{\mathrm{S}}(a_j \mid a_i, k_d) - \frac{1}{N_C} \qquad (4)$$

where N_C is the total number of models. The subtracted term $1/N_C$ ensures that non-informative transitions, i.e., transitions that are equally probable across all C, have a value of zero. The reason why (4) is not applied to novelty detection during learning is that it enforces a forced choice between the existing models. This leads to poor novelty detection performance since the probability mass of each transition across all models is always zero (note that activation values can be negative due to the subtraction of the constant). However, the normalization (4) has a significant impact on segmentation performance.

Now the activation A of each model c at each moment of time t is computed with

$$A(c,t) = \frac{1}{K} \sum_{d=1}^{K} P_c(X[t] \mid X[t-k_d], k_d) \qquad (5)$$

This provides a temporally local activation estimate for each model (Fig. 2). Then the activations are smoothed temporally using a simple moving average filtering in a 480 ms window. Only the most activated model for each moment of time is retained, leading to segmentation of the input into a discrete sequence of motif activations.

3.3. Segmentation of signal into activity epochs

The segmentation stage of the system studies the distribution of motifs and VQ-indices in a longer time window in order to discover epochs of internally coherent sensory activity that are assumed to correspond to different user contexts. During the process, a window of length L_W is moved in steps of L_S across the motif sequence and a histogram $\mathbf{h}_t$ of motif occurrences is computed for each window position. Each histogram is normalized to have a sum of one and cross-correlation matrix $\mathbf{C}_{\mathrm{motif}}$ is computed for all histograms:

$$\boldsymbol{C}(t_1, t_2) = \left(\boldsymbol{h}_{t_1} \cdot \boldsymbol{h}_{t_2}\right) / \left(\left\|\boldsymbol{h}_{t_1}\right\| \left\|\boldsymbol{h}_{t_2}\right\|\right) \qquad (6)$$

Now statistically coherent sections of the signal can be observed as square-formed plateaus of high correlation in the resulting matrix **C** (Fig. 3). The assumption is that the sensory context stays constant during such a coherent segment. In the experiments of this paper, values of L_W = 2000 (20 seconds) and L_S = 200 (2 seconds) were used. Cross-correlation matrix $\mathbf{C}_{vq}$ is also computed for the original VQ-sequence (distribution of VQ-indices), and elements of motif- and VQ-cross-correlation matrices are multiplied in order to obtain the final cross-correlation representation, i.e.

$$\mathbf{C}(t_1,t_2) = \mathbf{C}_{vq}(t_1,t_2)\mathbf{C}_{motif}(t_1,t_2). \quad (7)$$

In order to extract segments from **C**, a 2-dimensional filter is applied to the cross-correlation matrix (see [8]) that reacts strongly at the points in time where the coherence of signal changes suddenly. The filter is composed of one square region A of size d_1 x d_1 with its top-right corned placed against the diagonal of the matrix **C,** and of two identical triangles B_1 and B_2 with side lengths of d_2 that are next to the square and whose hypotenuses are also placed against the diagonal (Fig. 4). As the filter moves downwards along the diagonal, the means of cross-correlation matrix elements under the triangles B_1 and B_2 are subtracted from the mean of elements under the square A at each timed step.

$$s(t) = A(t) - B_1(t) - B_2(t) \quad (8)$$

This produces a signal $s(t)$ where deep valleys indicate segment boundary locations and valley depth indicates respective reliability of the segmentation. Simple peak/valley detection algorithm is then applied to extract temporal locations of segment boundaries.

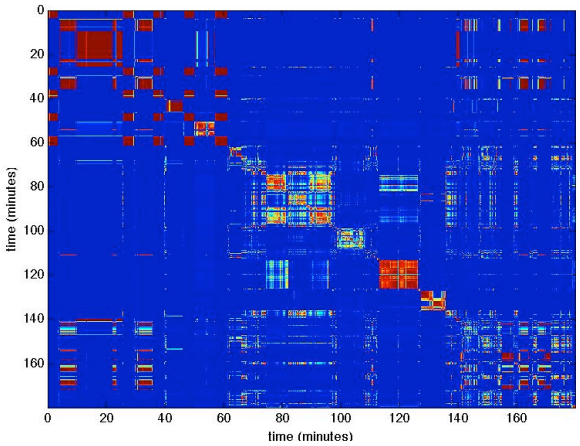

Fig. 3: Cross-correlation matrix **C** of motif histograms. Coherent epochs of sensory data can be seen as square sections along the diagonal.

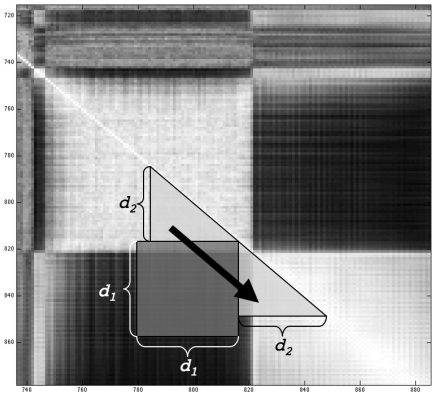

Fig. 4: 2D-filter used in the segmentation of the cross-correlation matrix **C**.

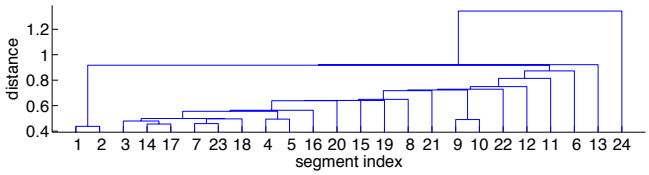

Fig. 5: An example clustering of discovered context segments.

3.4 Clustering of context segments

In order to cluster similar segments together, all segments are first represented by their characteristic features. These features include 1) the mean distribution $\mu_{vq,s}$ of VQ-indices over the segment, 2) the mean distribution $\mu_{m,s}$ of motifs over the segment, 3) variance of VQ indices over the segment, and finally 4) variance of motifs over the segment. In order to have equal weight for each feature, the feature vectors are all normalized separately to have a sum of one. Then the features are concatenated into one representative feature row-vector for each segment s:

$$f_s = [\mu_{vq,s}^T \mu_{m,s}^T \sigma_{vq,s}^{2\,T} \sigma_{m,s}^{2\,T}] \quad (9)$$

An agglomerative hierarchical cluster tree (Fig. 5) is constructed from the feature vectors by merging nearest feature vectors or clusters containing these vectors together at each step. Final clusters are defined by using a cutoff threshold d for cluster tree inconsistency, below of which all subtrees are considered as separate clusters, or as learned *context categories*. Inconsistency is defined as the ratio between edge weight and the average of other nearby edge weights.

3.5. Training of context category classifiers

Since the temporal boundaries between epochs of high-level contexts and corresponding category identities (cluster indices) are known, it is possible to use this labeling in association with the original sensory data in order to train a standard supervised classifier for each context category. This allows on-line detection of similar contexts in future data. In our work, we use computationally light Concept Matrix (CM) classifiers [9], but, e.g., standard HMM classifiers can be also used if sufficient computational resources are available.

4. EXPERIMENTS

A two-hour recording from one test subject was used to demonstrate the functionality of the system. For the given data, the SLVQ clustering produced N = 74 clusters for the MFCC features. Then the SLCM was used for learning with a novelty threshold of δ = 0.045, window length L = 1 second, window shift S = 1 second, and lags $\boldsymbol{k}$ = {1, 2, 3, 4, 5, 7, 9, 12, 15}, leading to the discovery of a total of 248 audio motifs from the VQ-data. Segmentation of the 2 hour VQ and motif data produced 23 segments, from which 19 clusters were obtained with cluster inconsistency threshold of d = 0.6.

Fig. 6 shows the result of the segmentation. Blue line denotes the annotated context (activities at the bottom and locations at the top) whereas red dashed lines show the detected segment boundaries and green bars at the bottom reflect the saliency of the boundary (higher bars correspond to higher saliency). As can be observed from the figure, the segment boundaries react mainly to the changes in user location, but sometimes user activity also causes changes in the discovered context. In terms of location,

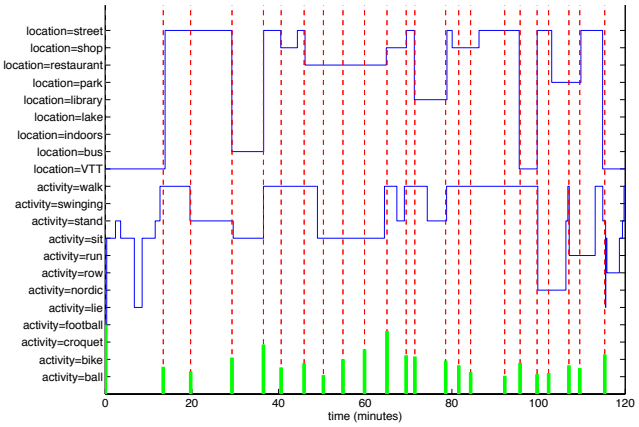

Fig. 6: Unsupervised context segmentation based on audio data. Blue line denotes the current context and red dashed lines show detected activity segment boundaries.

segmentation accuracy is relatively good with a mean deviation of $\sigma = 29.0$ s from reference boundaries and 5 insertions, mean purity of segments being 92%. For user activity, the mean deviation is $\sigma = 78.92$ s with -7 insertions (seven less boundaries were discovered than there are annotated boundaries). Table 1 shows the contents of discovered segments (left) and selectivity of the CM classifiers when they were validated with the same data (right).

Many of the differences between the segmentation and annotation can be understood by manually listening to the audio signals. For example, the over-segmented restaurant visit between 45 and 65 minutes actually consists of two distinct main stages with different auditory characteristics: queuing and operating at the cash desk and eating at the table. On the other hand, accurately segmented points typically consist of highly contrasting changes such as going indoors from the street or going to bus from the street.

5. CONCLUSIONS

A novel method for unsupervised discovery of high-level user contexts from low-level sensory data was presented in this work. The proposed system first discovers statistically significant short patterns, or motifs, from sensory data and then segments the signal into segments of different contexts by analyzing the distribution of patterns in the signal. These segments are then clustered into context categories based on their spectrotemporal similarities. Finally, a classifier is trained for each context category, making on-line recognition of previously encountered contexts possible.

The first experiments with recordings from various real-life situations show that the system can discover segments of audio activity that have high correspondence to manually annotated user locations. More experiments with multiple test subjects, independent test sets and additional sensors will be addressed in future work. For example, the performance of the on-line classifiers should be evaluated using a data set with independent test material in order to understand how well the system generalizes to similar but not exactly the same situations.

6. ACKNOWLEDGEMENTS

The author would like to thank Jukka P. Saarinen from Nokia NRC and Juha Pärkkä from VTT for making the Palantir dataset available.

Table 1: Contents of discovered context segments (left) and selectivity of learned on-line classifiers (right). Only 1-2 best matching classes are shown per token.

SEGMENTS			CLASSIFIER SELECTIVITY				
S	**%**	**ID**	**M**	**%**	**ID**	**%**	**ID**
1	95.8%	office	1	95.6%	street	4.4%	shop
2	93.2%	street	2	73.6%	shop	23.4%	street
3	99.7%	street	3	88.4%	street	8.5%	library
4	100.0%	bus	4	96.3%	street	3.7%	shop
5	96.0%	street	5	41.1%	shop	40.9%	street
6	70.5%	shop	6	100%	library		
7	95.2%	restaur.	7	93.2%	street	6.8%	office
8	100.0%	restaur.	8	97.0%	office	3.0%	street
9	100.0%	restaur.	9	50.8%	park	49.2%	street
10	96.8%	restaur.	10	89.7%	restaur.	10.3%	street
11	100.0%	shop	11	90.6%	park	9.4%	street
12	91.7%	street	12	100%	restaur.		
13	100.0%	library	13	100%	shop		
14	49.5%	shop	14	100%	office		
15	100.0%	shop	15	82.6%	bus	12.9%	street
16	75.1%	street	16	95.0%	shop	5.0%	restaur.
17	94.0%	street	17	89.5%	office	10.5%	street
18	100.0%	office	18	95.6%	street	4.5%	shop
19	98.8%	street	19	100%	restaur.		
20	84.4%	park					
21	100.0%	park					
22	86.5%	street					
23	100.0%	office					

7. REFERENCES

[1] Ermes M., Pärkkä J., Mäntyjärvi J., and Korhonen I., "Detection of Daily Activities and Sports With Wearable Sensors in Controlled and Uncontrolled Conditions," *IEEE Trans. Information Technology in Biomedicine*, 12(1), 2008.

[2] Pärkkä J., Ermes M., Korpipää P., Mäntyjärvi J., Peltola J., and Korhonen I., "Activity Classification Using Realistic Data From Wearable Sensors," *IEEE Transactions on Information Technology in Biomedicine*, Vol. 10, No. 1, 2006.

[3] Ma L., Milner B., and Smith D., "Acoustic Environment Classification," *ACM Transactions on Speech and Language Processing*, Vol. 3, No. 2, pp. 1-22, 2006.

[4] Clarkson B., and Pentland A., "Unsupervised Clustering of Ambulatory Audio and Video," *Proc. ICASSP'99*, pp. 3037-3040, Vol. 6, 1999.

[5] Krause A., Smailagic A., and Sewiorek D.. "Context-Aware Mobile Computing: Learning Context-Dependent Personal Preferences from a Wearable Sensor Array," *IEEE Trans. on Mobile Computing*, Vol. 5, No. 2, 2006.

[6] Räsänen O., Laine U.K., and Altosaar T., "Self-learning Vector Quantization for Pattern Discovery from Speech," *Proc. Interspeech '09*, Brighton, England, pp. 852-855, 2009.

[7] Räsänen O., "A computational model of word segmentation from continuous speech using transitional probabilities of atomic acoustic events," *Cognition*, Vol. 120, pp. 149-176, 2011.

[8] Räsänen O., Laine U., and Altosaar T. "Blind segmentation of speech using non-linear filtering methods," in Ipsic I. (Ed.): *Speech Technologies*, InTech Publishing, 2011.

[9] Räsänen O., and Laine U., "A method for noise-robust context-aware pattern discovery and recognition from categorical sequences," *Pattern Recognition*, Vol. 45, pp. 606-616, 2012.