LATENT SEMANTIC ANALYSIS IN SOUND EVENT DETECTION

Annamaria Mesaros [1], *Toni Heittola* [1], *Anssi Klapuri* [2]

[1] Department of Signal Processing
Tampere University of Technology
Korkeakoulunkatu 1, 33720, Tampere, Finland
email: annamaria.mesaros@tut.fi, toni.heittola@tut.fi

[2] Queen Mary University of London
Electronic Engineering & Computer Science
Mile End Road, London E1 4NS, UK
email: anssi.klapuri@elec.qmul.ac.uk

ABSTRACT

This paper presents the use of probabilistic latent semantic analysis (PLSA) for modeling co-occurrence of overlapping sound events in audio recordings from everyday audio environments such as office, street or shop. Co-occurrence of events is represented as the degree of their overlapping in a fixed length segment of polyphonic audio. In the training stage, PLSA is used to learn the relationships between individual events. In detection, the PLSA model continuously adjusts the probabilities of events according to the history of events detected so far. The event probabilities provided by the model are integrated into a sound event detection system that outputs a monophonic sequence of events. The model offers a very good representation of the data, having low perplexity on test recordings. Using PLSA for estimating prior probabilities of events provides an increase of event detection accuracy to 35%, compared to 30% for using uniform priors for the events. There are different levels of performance increase in different audio contexts, with few contexts showing significant improvement.

1. INTRODUCTION

A sound event can be described by a label that people would use to name a recognizable situation in a given environment. Such a label usually allows people to understand and associate the concept behind it with other known events. Usually they will be simply defining recognizable everyday sounds. Labels could be denoting human actions, like ”coughing”, ”sneezing”, denoting objects present on the scene, like ”car”, ”whistle”, or denoting the sound itself, like ”dog barking”. The free description given to such events may vary from one person to another, but the concept itself is generally understood by anyone. Automatic audio scene analysis tries to find such understandable labels, based on features extracted from or audio.

Event detection from audio has been studied for various applications, including audio scene recognition [1, 2, 3], analysis of video sound tracks [4, 5], acoustic event detection [6]. In other studies, individual sound events are considered to be characteristics of the audio scene and used for audio context recognition without actually naming the events themselves [7]. In general, these studies employ a rather limited set of events, small set of environments, and in many cases the sound events and audio examples are chosen so to minimize overlapping between different categories. In case of overlapping sound events, the annotation considers the most prominent one.

Our everyday environment is complex in sound events, and there is usually a high degree of overlapping between the sound events. As humans we can easily segregate sounds and identify multiple events at the same time, but for automatic recognition, the situation is challenging. There is little work that considers annotating specifically overlapping events, and even so, in [6] and [8], the output of the systems is a sequence of non-overlapping events. The assumption is that the detection result will contain at each time the most prominent sound event from the mixture, and the evaluation metric in [6] is developed for that situation. To our knowledge, there is no work that considers modeling and detecting overlapping events for event detection.

In [8] we presented a system for event detection in real life recordings, with an average 30% accuracy for a number of 61 event classes. The system is based on hidden Markov models (HMM) trained on mel-frequency cepstral coefficients (MFCC) and Viterbi decoding of best path through the HMM states. The attempt of including count-based priors in the decoding process did not bring any improvement to the performance.

In this paper we propose use of probabilistic latent semantic analysis (PLSA) to obtain the prior probabilities for the sound events. PLSA is widely used in text segmentation, for capturing long term relationships between words in text. Other uses include human action categorization in videos [9] and semantic concept annotation in audio [10]. In [10], the audio is first segmented into homogeneous segments and a vocabulary is constructed by clustering those segments. The probabilistic model is used to discover the topics existing in the audio clips and used in classification.

Our approach for learning the relationship between sound events will be using PLSA to model the co-occurrence of annotated events in the audio. We represent a fixed length segment of audio by an M-dimensional vector, where M is the number of events in the database. Each element $m(e)$ represents the percentage of the segment when the sound event e is active. We will use PLSA for discovering the underlying topics in the database and estimate prior probabilities for individual events e. These probabilities will be integrated in the decoding stage of the event detection system presented in [8].

The paper is organized as follows: Section 2 presents the details of the sound event detection system and the database. Section 3 presents the principles of probabilistic latent semantic analysis for modeling overlapping sound events, event probabilities and model evaluation. Section 4 presents experimental results: choosing parameters for the PLSA, estimated perplexities and integration of PLSA based event probabilities in the event detection system. Finally, Section 5 provides the conclusions of this work.

2. THE EVENT DETECTION SYSTEM

This paper presents a sound event detection system that uses probabilistic latent semantic analysis to provide prior probabilities to events in the detection stage. We consider the task of recognizing and locating sound events in recordings containing overlapping events. A previous version of the system was presented in [8], where the detection stage used uniform and count based priors. A better way of modeling the sound event probabilities should bring improvement to the performance.

The work presented in this paper uses audio recorded in different environments, with the aim of detecting predetermined classes of sound events. The recordings are made in ten different audio contexts: basketball game, beach, inside a bus, inside a car, office, hallway, restaurant, shop, street and track&field stadium, and contain highly overlapping events. The data consists of a total of 103 recordings. There are 8 to 14 recordings per context, with lengths of 10 to 30 minutes, having a total duration of 1133 minutes.

The recordings are annotated to indicate presence/absence of sound events as a function of time. The annotations mark start and end times of sound events belonging to 61 classes; the annotated events are overlapping, with no limit on the number of events that

Table 1: Number of events annotated in each context

basketball	990	beach	738
bus	1729	car	582
office	1220	hallway	822
restaurant	780	shop	1797
street	827	tracknfield	793

can be active at the same time. In this sense, the data is *polyphonic*. Within each context there are from 9 to 16 annotated event classes. Some events are specific to contexts, for example ”car door”, while others are present in more or in all the contexts, like for example ”speech” or ”footsteps”. Some events may be very short (”beep”), others may extend for the whole duration of the file (”ventilation noise”). The most quiet environment is inside of car, with a number of 582 total annotated sound events, and the most complex is the shop with almost 1800 annotated events. Table 1 presents information about the number of events in each context.

The acoustic models of the event classes are constructed using hidden Markov models (HMM) and mel-frequency cepstral coefficients (MFCCs) as features. The features for each annotated event instance were calculated directly from the polyphonic mixture. In the case when more events appear simultaneously, the same part of the audio (therefore the same observation vectors) was assigned to all the event classes present in that segment. The observations calculated for individual events were used to construct the models for each class.

The output of the system is a *monophonic* sequence containing the most prominent events. This is obtained using the Viterbi algorithm, performing both recognition and segmentation of the audio, based on the 61 models. The event HMMs are connected into a network. The PLSA will be integrated at the transitions from one model to the other within the HMM network. At each step, the algorithm will choose the new model based on inter-model transition probabilities provided by event priors based on PLSA.

3. PROBABILISTIC LATENT SEMANTIC ANALYSIS

3.1 General concept

PLSA is a statistical technique for the analysis of co-occurrence data, based on a mixture decomposition derived from a latent class model. The most common application for latent semantic analysis is analysis and retrieval of text documents. For a collection of text documents $D = d_1,..d_N$ and a vocabulary $W = w_1,..w_M$, by ignoring the order in which words occur, the documents are summarized as a co-occurrence matrix with terms $c(w_m,d_n)$ representing how many times word w_m appeared in document d_n.

PLSA models the probability of each co-occurrence as a mixture of conditionally independent multinomial distributions. The latent variable $z \in Z = z_1..z_K$ in PLSA is called an aspect model and in the text it represents topics. The joint probability model over the documents and words is defined by

$$P(d,w) = P(d)\sum_{z\in Z} P(w|z)P(z|d) \tag{1}$$

The model can be equivalently parametrized by:

$$P(d,w) = \sum_{z\in Z} P(z)P(d|z)P(w|z) \tag{2}$$

The first formulation is the asymmetric formulation: for each document d, a latent class is chosen conditionally to the document according to $P(z|d)$, and a word is generated from that class according to $P(w|z)$ (Figure 1.a). In the symmetric formulation (Figure 1.b), w and d are both generated from the latent class z in similar ways, using the conditional probabilities $P(d|z)$ and $P(w|z)$.

The modeling assumption is that the conditional distributions $P(w|d)$ are approximated by a combination of factors $P(w|z)$, with the mixing weights $P(z|d)$ uniquely defining a point in the latent space [11]. The model is obtained by expectation maximization algorithm [12]. Each document is viewed as a mixture of topics, and each topic is represented by a combination of the words. The number of topics is smaller than the size of the vocabulary or the number of documents, leading to a robust estimation of the two components.

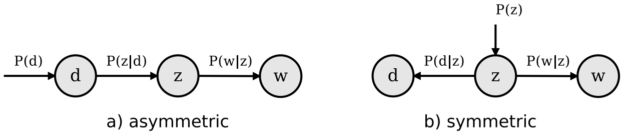


Figure 1: Graphical model representation of the two formulations of PLSA: (a) asymmetric, (b) symmetric

3.2 PLSA for modeling overlapping events

The concept of PLSA can be extended to model co-occurrence of any pair of discrete variables. In our data, the sound events take the role of words, and the documents are segments of audio. The main difference between events in a segment and words in a text is that the events can be overlapping, therefore appearing simultaneously, not strictly as a sequence. We will consider co-occurrence of events in fixed length audio segments.

Within a segment, we estimate for each event the percentage of its active state. Each audio file will be represented by a matrix in which rows represent the sound events and individual columns represent the active time of each event in consecutive time segments. The process is illustrated in Figure 2.

We include a universal background model event (UBM) to be present all the time in all the segments. This model represents the overall properties of the data; it is trained on all the data and its role in decoding is to capture the regions when no events of interest are detected.

For the audio data, the count matrix C represents the event-by-segment presence. We will use the following notations: e for events, z for topics, s for segments. The conditional probabilities between events and topics and topics and segments respectively are represented as a basis matrix B (event-by-topic) containing $p(e|z)$ and a gain matrix G (topic-by-segment) containing $p(z|s)$. The matrices B and G are obtained using the expectation maximization algorithm. Their formulation is represented in Figure 3.

3.3 Obtaining the event probabilities from PLSA

The dependencies between events are modeled by topics obtained using PLSA. The probabilities of events at each step in the decoding will be calculated depending on a local topic defined by the events that were recognized so far. We call this a *history*. The history is obtained via Viterbi decoding and is represented as sequence of non-overlapping events, due to the nature of the event detection system. The next event in this sequence will be decoded by taking into account relationships between the history and the possible events.

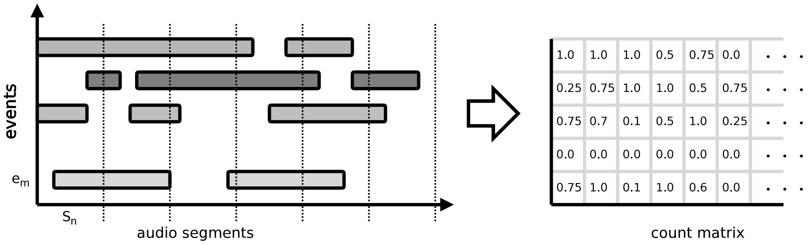
The probability of events given the history is obtained by:

$$p(e|h) = \sum_{z} p(e|z)p(z|h) \tag{3}$$

We will use a history of length L seconds and represent it as the set of events that were recognized in the previous L seconds, $h = \{e\}_{t-L}^{t}$.

The term $p(e|z)$ is obtained directly in the model. The term $p(z|h)$ is the distribution of topics for history h. We will use the notation $p(z|\{e\}_{t-L}^{t})$ for it, to explicitly include the information about the history length. We calculate it as:

$$p(z|\{e\}_{t-L}^{t}) = \frac{p(\{e\}_{t-L}^{t}|z)p(z)}{p(\{e\}_{t-L}^{t})} \tag{4}$$



1.0	1.0	1.0	0.5	0.75	0.0	. . .
0.25	0.75	1.0	1.0	0.5	0.75	. . .
0.75	0.7	0.1	0.5	1.0	0.25	. . .
0.0	0.0	0.0	0.0	0.0	0.0	. . .
0.75	1.0	0.1	1.0	0.6	0.0	. . .

Figure 2: Forming the events-to-segments count matrix based on the presence if audio events in each audio segment of a given length

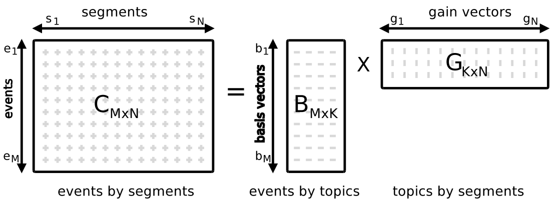


Figure 3: Representation of the event co-occurrences in the PLSA framework using a basis matrix B that represents conditional probabilities of events by topics and a gain matrix G that represents the conditional probabilities of topics by segments

The first term in the nominator represents the probability of the history. For simplification, we assume independence of events, and will calculate the probability of the history as the product of the individual probabilities of events that are present in the history, normalized by taking into account the cardinality of the set, $|\{e\}_{t-L}^{t}|$.

$$p(\{e\}_{t-L}^{t}|z) \approx \left[\prod_{z} p(e|z)_{e\in\{e\}_{t-L}^{t}}\right]^{\frac{1}{|\{e\}_{t-L}^{t}|}} \qquad (5)$$

The second term in the nominator is the probability of the topics and is calculated by summing over the N segments s in the gain matrix G.

$$p(z) = \sum_{s} p(z|s)p(s) = \frac{1}{N}\sum p(z|s) \qquad (6)$$

With these, we can now calculate the probabilities of events given the recent history of L seconds:

$$p(e|\{e\}_{t-L}^{t}) = \sum p(e|z)p(z|\{e\}_{t-L}^{t}) \qquad (7)$$

The beginning of the decoding presents a special situation. The first calculation of priors has no history available, and the probabilities have to be calculated based on the distribution of topics observed in the training data:

$$p(e) = \sum p(e|z)p(z) \qquad (8)$$

This forms an initial estimate for the priors and is used until we accumulate enough history for making better predictions. When using history, the event probabilities are being adapted continuously during the decoding.

3.4 Evaluation of the PLSA model

A model q based on a training sample that was drawn from the unknown distribution p is evaluated using the cross-entropy of the models:

$$H(p,q) = -\sum_{e} \tilde{p}(e)log_2 q(e) = -\frac{1}{S}\sum_{e} log_2 q(e) \qquad (9)$$

where $q(e)$ is the probability predicted by the model q, S is the length of the sequence and $\tilde{p}(e)$ is the observed distribution of the test sample, $1/S$ because each sample appeared once in the test data of size S.

In the evaluation of language models for automatic speech recognition, a common evaluation measure derived from entropy is perplexity, $P = 2^{H(p)}$, where $H(p)$ is the entropy of the discrete probability distribution p. Perplexity can be interpreted as how confused is the model on the test data, having to choose uniformly and independently among P possibilities. Better models q of the unknown distribution p will tend to assign higher probabilities $q(e)$ to the test events, therefore they have lower perplexity.

We modify the interpretation of perplexity to our data. The output of the detection system is a monophonic sequence, while the annotation contains multiple events. This means there are a number of events that represent a correct output: any of the events that are marked active in the annotation. As an example, consider throwing a fair die (uniform distribution over 6 discrete events) to obtain a sequence of numbers. If we evaluate perplexity for modeling the value of each event, the perplexity of the uniform distribution model is 6. The model chooses among 6 possibilities. But if we evaluate perplexity for the quality of odd or even in the sequence, there are 3 possible correct answers at each throw, and the perplexity of the same model will be 2.

To calculate the entropy of the model we will use the total probability of the set of correct events:

$$q(x) = \sum_{e\in\{e\}_{correct}} q(e) \qquad (10)$$

In this way, the entropy represents the amount of the probability mass that is assigned by the model to the correct events. The correct events are the ones that are marked as being active in the considered audio segment.

4. EXPERIMENTAL RESULTS

We evaluate the perplexity of the constructed model on data similar to the training. The universal background model has to be part of the PLSA model in order to have similar properties as the event classes in the decoding process; when evaluating event detection we consider only the classes of interest and do not count the UBM.

The history used for calculating the event priors can contain errors, because it is the output of the events detection system and it will naturally have some errors. To counteract the effect of class confusions, we create noisy training data for the PLSA. We use a classifier for isolated sound events [8] to classify the training data. The resulting confusion matrix $S_{(M\times M)}$ is used to smooth the clean annotations by multiplying the count matrix $C_{(M\times N)}$ with it:

$$C^{noisy}_{(M\times N)} = S_{(M\times M)} \times C_{(M\times N)}. \qquad (11)$$

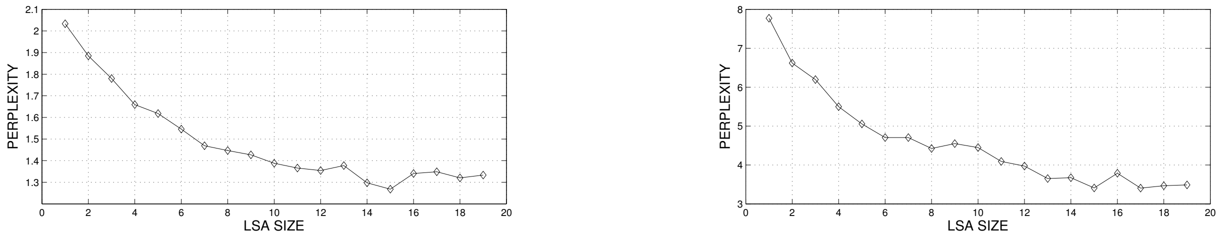


Figure 4: Perplexity of PLSA model using clean and noisy training data for different size PLSA

Table 2: Perplexity of PLSA model for different size semantic space. Test and train data are ground truth annotation

PLSA	training segment length in seconds							
size	5	10	15	30	45	60	90	120
6	1.7	1.7	1.7	1.6	1.5	1.5	1.5	1.5
8	1.6	1.5	1.5	1.5	1.4	1.4	1.4	1.4
10	1.5	1.4	1.4	1.4	1.4	1.4	1.4	1.4
12	1.4	1.4	1.4	**1.3**	1.3	1.3	1.3	1.3
14	1.5	1.4	1.4	1.3	1.3	1.3	1.3	1.3
16	1.4	1.4	1.4	1.3	1.3	1.2	1.3	1.3
18	1.4	1.4	1.4	1.3	1.3	1.3	1.3	1.3
20	1.4	1.3	1.4	1.3	1.2	1.3	1.2	1.3

This way the predictions will be made based on counts of acoustic confusions between the considered event classes.

4.1 Perplexity of the PLSA model

The test data for evaluating perplexity consists in ground truth annotations marked for the presence/absence of an acoustic event in each segment. They are similar with the PLSA training data, but used only as binary indicators, stating the event is or is not active in the tested segment. The entropy will be calculated as presented in the previous section, by considering the probability mass assigned by the model to all the events marked active in the tested segment.

We need to choose the dimensionality of the semantic space and the size of the segments representing the documents for the training. We will make the choice based on the perplexity evaluation. The perplexity of a uniform distribution model on the test data was calculated for comparison. Its value is between 9 and 10, consistent with an average of 6 overlapping events in each time segment ($62/6 \approx 10$).

We calculated average perplexity in 5 folds, for different values of training segment length and PLSA size. We used equal lengths for training segments and history. They can have different lengths, but we chose to have them the same length. The results are presented in Table 2. We choose to have the highest reduction in dimensionality that offers a good performance, and we settle for a PLSA size of 12. To allow correcting errors in prediction based on erroneously detected events, we would choose a short length history. According to the numbers in Table 2, we choose a segment/history length of 30 seconds. To reiterate, perplexity can be interpreted as the number of possibilities from which the model has to choose uniformly, and in this case it is less than 2.

The values of the perplexity have a similar trend when the training data is smoothed using the confusion matrix. Figure 4 presents perplexity of the PLSA model when trained using clean and noisy data, for a PLSA size varying from 2 to 20. In both cases a size of 12 looks as acceptable choice, offering good dimensionality reduction and small enough perplexity.

We compare the perplexity of the PLSA with history to the possibility of using only the initial PLSA estimates. These do not depend on history, their values are constant throughout the whole detection step, and are calculated directly from the training data according to equation 8. Table 3 presents the resulting perplexity values for different methods of assigning probabilities to events: uniform distribution of the event classes, PLSA with size 12 and no history, PLSA with size 12 and history length 30 seconds. Results are presented for the PLSA models trained using both clean and noisy data.

Table 3: Perplexity of uniform distribution, PLSA without history and PLSA with a 30 seconds history; mean and standard deviation of perplexity values

uniform distrib		11.8 (4.0)
clean training data	PLSA no history	2.2 (0.3)
	PLSA with 30s history	1.3 (0.3)
noisy training data	PLSA no history	8.8 (6.5)
	PLSA with 30s history	3.9 (3.6)

When using the constant priors and no history in the perplexity calculation, it appears that 50% of the probability mass is assigned to the correct events – this seems to be already a very good model. In reality, a large amount of this probability mass is explained by the universal background model, which gets a very strong prior probability, being active in all segments. The probability mass assigned to the events of interest is around 20%. The use of history reduces the perplexity to almost half of this, still with a quite large amount of probability mass explained by the UBM. The multiplication with the confusion matrix smooths out the priors and reduces the probability mass associated with the UBM.

4.2 Sound event detection results

The accuracy of the detection is measured in terms of balanced F-score between precision and recall. This metric has been developed for acoustic event detection in case of polyphonic annotation and monophonic output [6]. The precision represents the number of events correctly detected. The output is considered correct if there exists at least one annotated event whose temporal center is situated between the timestamps of the system output, and the annotated label and system output are the same, or if the temporal center of the system output lies between the timestamps of at least one annotated event and the annotated label and system output are the same. The recall represents the number of events from the annotation that were correctly detected. The annotated sound event is considered correctly detected if there exists at least one system output whose temporal center is situated between the timestamps of annotated sound event and the labels are the same, or if the temporal center of the annotated sound event lies between the timestamps of at least one system output and the labels are the same.

We compare the performance of the detection system that incorporates PLSA derived priors with the baseline system from [8]. Table 4 presents event detection results for the baseline system and for using event priors based on PLSA with and without history, and event priors based on PLSA trained on the noisy data (ground truth annotation smoothed with confusion matrix).

Table 4: Event detection accuracy using PLSA based priors

baseline	PLSA	PLSA	PLSA$_{noisy\_train}$
system	no history	with history	with history
30.1%	32.5%	32.6%	34.9%

The predictions of the PLSA model are very good, and the perplexity reduction reflects this well. The performance of the event detection system though does not improve very much. There could be a number of reasons for this: erroneous predictions because of incorrect history, a poor performance of the acoustic models, or the limitation of outputting only one event when there are more events overlapping.

If the history contains events that are not correct, this will affect the probabilities estimated for the events when decoding the next few events. To eliminate this possibility, we used ground truth as history and tested the event detection system. The detailed results can be found in Table 5. The average performance in these conditions was 36.3%, not significantly higher than the 34.9 % performance of the system that incorporates PLSA based on noisy data. A closer analysis of the results reveals big differences in the influence of the PLSA based priors to detecting events in different audio contexts. Contexts with very specific type of events like basketball and track&field benefit mostly of the ground truth (GT) history. Contexts like bus, hallway, street get some improvement. The car is one example of very sparse environment, for which the perplexities were a lot larger than for the other contexts, and the detection results with PLSA are lower than the ones of the baseline system.

The acoustic models are not powerful enough to offer higher performance. Training acoustic models from complex polyphonic data is challenging, and some refining is needed in selecting the audio used for each class, in order to obtain better performing models. In the same time, the performance is limited by the fact that we decode a monophonic sequence, while the annotations contain overlapping events. For an average polyphony of 6 simultaneous events, in the ideal case of decoding a correct event at each time, the maximum recall is 1/6, and the precision would be 1, leading to average maximum performance of about 28% (33% for 5, 40% for 4 simultaneously active events). The only way to overcome this problem is to find methods for outputting multiple overlapping events.

In our future work we will use sound source separation techniques both in training and testing. In training, separation of sound sources will help in improving the acoustic models, while in testing, the method will provide a way to detect several overlapping sound events.

5. CONCLUSIONS

This paper presented the use of probabilistic latent semantic analysis for estimating priors in a sound event detection system. The performance of the event detection system is limited both by the acoustic models and by decoding a monophonic output, and these aspects need to be addressed in the future. The proposed method based on PLSA solves the problem of including prior information about event classes by estimating prior probabilities for individual events in the event detection.

PLSA was used to model co-occurrence of events based on the degree of their overlapping during a fixed length segment. The modeling offers reliable priors for the events, based on the set of events present in the decoded history. Errors in the decoded sequence can influence the probability estimations for events to be decoded next. To counteract this, the annotation count matrix was smoothed using the confusion matrix of the training data. In this situation, the PLSA model has a perplexity of 3.9, favoring approximately 4 equally probable events. In comparison with a baseline system that uses uniform distribution for event priors, the average increase in performance is from 30% to 35% detection accuracy.

Table 5: Context-wise event detection accuracy

Context	PLSA$_{noisy\_train}$	PLSA$_{noisy\_train}$
	with history	with GT history
Basketball	48.4	55.0
Beach	32.4	31.9
Bus	32.6	33.5
Car	33.3	31.9
Hallway	37.1	38.5
Office	57.5	55.6
Restaurant	21.3	25.0
Shop	26.8	26.9
Street	20.0	22.0
Track&Field	38.2	42.6
Overall	34.9	36.3

REFERENCES

[1] C. Clavel, T. Ehrette, and G. Richard, “Events detection for an audio-based surveillance system,” in *Proc. of the 2005 IEEE International Conference on Multimedia and Expo, ICME 2005*, 2005, pp. 1306–1309.

[2] A. Härmä, M. F. McKinney, and J. Skowronek, “Automatic surveillance of the acoustic activity in our living environment,” in *Proc. of IEEE International Conference on Multimedia and Expo*, Los Alamitos, CA, USA, 2005, p. 4 pp., IEEE Computer Society.

[3] A.J. Eronen, V.T. Peltonen, J.T. Tuomi, A.P. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, “Audio-based context recognition,” *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 321–329, Jan. 2006.

[4] R. Cai, L. Lu, A. Hanjalic, H-J. Zhang, and L-H. Cai, “A flexible framework for key audio effects detection and auditory context inference,” *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 3, pp. 1026–1039, 2006.

[5] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, “Audio keywords generation for sports video analysis,” *ACM Trans. on Multimedia Computing, Communications and Applications*, vol. 4, no. 2, pp. 1–23, 2008.

[6] A. Temko, R. Malkin, C. Zieger, D. Macho, C. Nadeu, and M. Omologo, “CLEAR evaluation of acoustic event detection and classification systems,” in *Multimodal Technologies for Perception of Humans*. 2007, Lecture Notes in Computer Science.

[7] S. Chu, S. Narayanan, and C-C. Jay Kuo, “Environmental sound recognition with time-frequency audio features,” *IEEE Trans. on Speech, Audio, and Language Processing*, vol. 17, no. 6, pp. 1142–1158, 2009.

[8] A. Mesaros, T. Heittola, T. Virtanen, and A. Eronen, “Acoustic events detection in real life recordings,” in *Proc. of the 2010 European Signal Processing Conference (EUSIPCO-2010)*, 2010, pp. 1267–1271.

[9] J. C. Niebles, H. Wang, and L. Fei-Fei, “Unsupervised learning of human action categories using spatial-temporal words,” *International Journal of Computer Vision*, 2008.

[10] Y. Peng, Z. Lu, and J. Xiao, “Semantic concept annotation based on audio plsa model,” *ACM Multimedia*, pp. 841–844, 2009.

[11] T. Hofmann, “Probabilistic latent semantic analysis,” in *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI’99)*, 1999, pp. 289–296.

[12] A.P. Dempster, N.M. Laird, and D.B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.