IJCV manuscript No.
(will be inserted by the editor)

A Sequential Topic Model for Mining Recurrent Activities from Long Term Video Logs.

Jagannadan Varadarajan · Rémi Emonet · Jean-Marc Odobez

Received: date / Accepted: date

Abstract This paper introduces a novel probabilistic activity modeling approach that mines recurrent sequential patterns called *motifs* from documents given as word×time count matrices (*e.g.,* videos). In this model, documents are represented as a mixture of sequential activity patterns (our motifs) where the mixing weights are defined by the motif starting time occurrences. The novelties are multi fold. First, unlike previous approaches where topics modeled only the co-occurrence of words at a given time instant, our motifs model the co-occurrence and temporal order in which the words occur within a temporal window. Second, unlike traditional Dynamic Bayesian Networks (DBN), our model accounts for the important case where activities occur concurrently in the video (but not necessarily in synchrony), *i.e.,* the advent of activity motifs can overlap. The learning of the motifs in these difficult situations is made possible thanks to the introduction of latent variables representing the activity starting times, enabling us to implicitly align the occurrences of the same pattern during the joint inference of the motifs and their starting times. As a third novelty, we propose a general method that favors the recovery of sparse distributions, a highly desirable property in many topic model applications, by adding simple regularization constraints on the searched distributions to the data likelihood optimization criteria. We substantiate our claims with experiments on synthetic data to demonstrate the algorithm behavior, and on four video datasets with significant variations in their activity content obtained from static cameras. We observe that using low-level motion features from videos, our algorithm is able to capture sequential patterns that implicitly represent typical trajectories of scene objects.

Keywords Unsupervised · Latent sequential patterns · Topic models · PLSA · Video surveillance · Activity analysis

Authors gratefully acknowledge the support of the Swiss National Science Foundation (Project: FNS-198, HAI) and of the European Union through its 7th framework program (Integrated project VANAHEIM(248907, and Network of Excellence PASCAL2).

Jagannadan Varadarajan, Rémi Emonet, Jean-Marc Odobez
Idiap Research Institute, Martigny, Switzerland
École Polytechnique Fédéral de Lausanne, Switzerland
E-mail: vjagann@idiap.ch, remonet@idiap.ch, odobez@idiap.ch.

(a) MIT data (b) Far field (c) Traffic Junction (d) Metro Station

Fig. 1: Surveillance scenes

1 Introduction

Immense progress in sensor and communication technologies has led to the development of devices and systems recording multiple facets of daily human activities. This has resulted in an increasing interest for research on the design of algorithms capable of inferring meaningful human behavioral patterns from data logs captured through sensors, simultaneously leading to new application opportunities. The surveillance domain is a typical example. From videos of scenes such as those illustrated in Fig. 1 that are obtained from static cameras, one would like to automatically discover the typical activity patterns, when they start or end, or predict an object's behavior. Such information can be useful in its own right, *e.g.,* to better understand the scene content and its dynamics, or for higher semantic level analysis. For instance it would be useful to define the data-driven real camera activities, to provide context for other tasks (*e.g.,* object tracking) or to spot abnormal situations which could for instance be used to automatically select the relevant camera streams to be displayed in control rooms of public spaces monitored by hundreds of cameras.

Most activity analysis approaches are object-centered where objects are first detected, then tracked and their trajectories used for further analysis [33, 26, 48, 16]. Tracking-based approaches provide direct object-level semantic interpretation, but are sensitive to occlusion and tracking errors especially in crowded or complex scenes where multiple activities occur simultaneously, and usually require substantial computational power. Thus, as an alternative, researchers have successfully investigated algorithms relying on low-level features like optical flow that can readily be extracted from the video stream to perform activity analysis tasks such as action recognition or abnormality detection [6, 51, 52, 25, 30].

In visual surveillance, unsupervised methods are preferred since, due to the huge inflow of data, obtaining annotations is laborious and error prone. Such unsupervised techniques, relying on simple features like location and motion were proposed to perform scene activity analysis and abnormal event detection in [56, 51, 52, 25]. Note that low-level visual features like color histograms [42] and optical flow [11] along with statistical models were earlier shown to be effective for video content analysis, indexing and retrieval. Some conventional approaches involved a clustering framework on top of the extracted optical flow features. For instance, [52] relied on diffusion maps to capture regular propagation of activities, while [30] exploited a hierarchical clustering approach relying on KL-divergence based comparison measure. Among unsupervised techniques, however, topic models originally proposed for text processing [17, 5] like Probabilistic Latent Semantic

Analysis (PLSA) [17] or Latent Dirichlet Allocation (LDA) [5] have emerged as a tool with tremendous potential due to their ability to capture dominant themes in large data collections.

Topic models were first applied in vision for tasks like scene [29], object [32] and action [28, 49] recognition. More recently, they have been shown to be successful in discovering scene level activities as dominant spatio-temporal word patterns by considering quantized spatio-temporal visual features as words and short video clips as documents. For instance, [46] used a hierarchical variant of LDA to extract atomic actions and interactions in traffic scenes, while [23] relied on hierarchical PLSA to identify abnormal activities and repetitive cycles. Activity based scene segmentation and a detailed study of various abnormality measures in this modeling context is done in [41].

Although such approaches are able to discover scene activities, the actual modeling of temporal information remains an important challenge. By relying only on the analysis of unordered word counts (due to the bag-of-words approach) within a time window, most topic models fail to represent the sequential nature of activities, although activities are often temporally ordered. For example, in traffic scenes, people wait at zebra crossings until all vehicles have moved away before crossing the road, giving rise to a temporally localized and ordered set of visual features. Using a "static" distribution over features to represent this activity may be concise but not complete, as it does not allow us to distinguish it from an abnormal situation where a person crosses the road while vehicles are still moving.

In this paper, we propose an unsupervised approach based on a novel graphical topic model called *Probabilistic Latent Sequential Motifs (PLSM)*, for discovering dominant sequential activity patterns called *motifs* from sensor data logs represented by word×time counts or *temporal documents*. In this context, the main contributions of our paper are:

- a model that learns sequential activity patterns called motifs - that not only capture the co-occurrence of words in a temporal window, but also the *temporal order* in which the words occur within this window;
- a model that accounts for the important case where *temporal activities occur concurrently* in the document (but not necessarily in synchrony), *i.e.,* several activities might be going on at a given time instant;
- an estimation scheme that performs *joint inference of the motifs and their starting times*, allowing us to implicitly align the occurrences of the same pattern during learning;
- a simple regularization scheme that encourages the *recovery of sparse distributions* in topic models, a highly desirable property in practice, which can be used with most topic models (*e.g.,* PLSA, LDA).

The behavior of the algorithm is validated on synthetic data, and its ability to capture sequential activities is demonstrated on four different videos of public scenes captured using static cameras. The datasets come from state of the art papers. We believe that our contribution is quite fundamental and relevant to a variety of applications where sequential motifs ought to be discovered out of time series arising from multiple activities. This paper improves substantially on our initial work [38, 40], from both a modeling (sparsity approach, MAP optimization and model selection) and experimental perspective (more thorough synthetic experiments, exhaustive experiments on traffic videos involving vehicles and pedestrians

and metro station views containing less structured movement of people, numerical evaluation and comparison with state-of-the-art algorithms on a prediction task).

The plan of the paper is as follows. In section 2, we analyze the state-of-art and compare it with our approach. Section 3 introduces our PLSM model with details, including the inference procedure. Experiments on synthetic data are first conducted in section 4 to effectively demonstrate various aspects of the model. The application of the PLSM model to the extraction of recurring activities in surveillance videos is explained in section 5, which includes experiments on three different video datasets from state of the art papers and a video dataset from a crowded metro station. The captured PLSM motifs are shown and discussed in section 6, with quantitative experiments on an activity prediction task and on a comparison with ground truth labeled data. We discuss some limitations of the model and areas of future work in section 7 and finally provide the conclusions in section 8.

2 Related Work

Our work pertains to three main issues: the modeling of activities with topic models, the discovery of temporal motifs from time series, and the learning of sparse distributions. In this Section, we briefly review the prior works conducted along these aspects and contrast them with our work.

2.1 Temporal modeling with topic models

Topic models stem from text analysis and were designed to handle large collections of documents containing unordered words. Recently, however, several approaches have been proposed to include sequential information in the modeling. This was done either to represent single word sequences [43,15], or at a higher level, by modeling the dynamics of topic distributions over time [4,45,14]. For instance, [43] introduced word bigram statistics within a LDA-style model to represent topic-dependent Markov dependencies in word sequences, while in the Topic over Time method of [47], topics defined as distributions over words and time were used in a LDA model to discover topical trends over the given period of the corpus.

Many of these temporal models have been adapted for activity analysis. For instance, [18] introduced a Markov chain on scene level behaviors, but each behavior is still considered as a mixture of unordered (activity) words. More recently, [22] used the HDP-HMM paradigm (*i.e.,* Hierarchical Dirichlet Process, HDP, and Hidden Markov Model HMM) of [35] to identify multiple temporal topics and scene level rules. Unfortunately, for all four tested scenes only a single HMM model was discovered in practice, meaning that temporal ordering was concretely modeled at the global scene level using a set of static activity distributions, similar to what was done in [18,19]. Another attempt was made in [24], which modeled topics as feature$\times$time temporal patterns, trained from video clip documents where the timestamps of the feature occurrences relative to the start of the clip were added to the feature. However, in this approach, the same activity has different word representations depending on its temporal occurrence within the clip, which prevents the learning of consistent topics from the regularly sampled video clip documents.

To solve this issue of activity alignment with respect to the clip start, in [12], manually segmented clips were used so that the start and end of each clip coincided with the traffic signal cycles present in the scene. This method has two drawbacks: firstly, only topics synchronized with respect to the cycle start can be discovered. Secondly, such a manual segmentation is time consuming and tedious. Our model addresses both these issues.

Our method is fundamentally different from all of the above approaches. The novelties are that i) the estimated patterns are not merely defined as static distributions over words but also incorporate the temporal order in which words occur; ii) the approach handles data resulting from the temporal overlap between several activities; and iii) the model allows us to estimate the starting times of the activity patterns automatically.

2.2 Motifs from time series

An alternative view of activity discovery is that videos are time-series data and the various activity patterns are temporal motifs occurring in the multivariate time series. In this view, there has been some work on unsupervised activity discovery, which typically relied on Hidden Markov Model (HMM) and its variants to perform jointly a temporal segmentation of the time series, and learning (and identification) of activity patterns from feature vectors. For instance, in [55] activities of individual people are clustered jointly into meeting actions using a semi-supervised layered-HMM. However, these methods assume that the entire feature vector at a given time instant corresponds to a single activity. This precludes their use in our case where multiple activities can overlap without any particular order or synchronization, resulting in a mixing of their respective features at a given time instant.

Motif discovery from time series has also been an active research area in fields as diverse as medicine, entertainment, biology, finance and weather prediction to name a few [27]. However, these methods only solve scenarios where either one or several of the following restrictions hold: there is prior knowledge about the number of patterns or the patterns themselves [21]; the data is univariate; and most importantly they assume that there is only a single pattern occurring at any time instant [34]. To the best of our knowledge, our method is one of the first attempts in discovering motifs from time series where the motifs can overlap in time.

2.3 Learning sparse distributions

One common issue in non-parametric topic models is that distributions are often loosely constrained, resulting in non-sparse process representations which are often not desirable in practice. Similar to the *sparse coding* representational scheme [54], what we seek are distributions where most of the elements in a vector are zero, while few elements are significantly different from zero. For instance, in PLSA, one would like each document d to be represented by a few topics z with high weights $P(z|d)$, or each topic $P(w|z)$ to be represented by only a few words with high probability. But in practice, nothing guides the learning procedure towards

such a goal. The same applies to LDA models despite the presence of priors on the multinomial $P(z|d)$ [44].

Approaches to this problem have been proposed in areas related to topic models. In Non-negative Matrix Factorization (NMF), a non-probabilistic model close to PLSA, [20] proposed to set and enforce through constrained optimization, an *a priori* sparsity level defined by a relationship between the L1 and L2 norm of the matrices to be learned. Very recently, [44] introduced a model that decouples the need for sparsity and the smoothing effect of the Dirichlet prior in HDP, by introducing explicit selector variables determining which terms appear in a topic. A more complex model called focused topic model of [50] similarly addresses sparsity for hierarchical topic models but relies on an Indian Buffet Process to impose sparse yet flexible document-topic distributions.

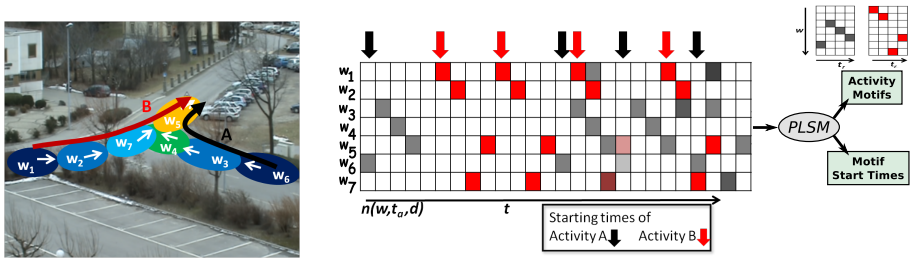
To address the sparsity issue, we propose an alternative approach. The main idea is to guide the learning process towards sparser (more peaky) distributions characterized by smaller entropy. We achieve this by adding a regularization constraint in the EM optimization procedure that favors lower entropy distributions by maximizing the Kullback-Leibler divergence between the distribution to be learned and the uniform distribution (maximum entropy). This results in a simple procedure that can be applied to most topic models where a sparsity constraint on the distribution is desirable.

3 Probabilistic Latent Sequential Motif Model

In this section, we first illustrate the model principle for temporal activity discovery in videos, and then introduce our notation and an overview of the model. In the second part, we provide a more detailed description of the generative process of the model, the EM steps derived to infer the model parameters, the employed sparsity constraint, the exploitation of priors, and model selection.

3.1 Illustration of the PLSM model for videos

Before formalizing our approach, we illustrate the activity discovery algorithm on a simple video case, as shown in Fig. 2. Assume that in this scene, only two activities can occur, and that we have a vocabulary of $N_w = 7$ words, where each word characterizes the motion activity happening in some local regions (the colored blobs in Fig. 2(a)). The method to automatically define these scene specific words is described in Section 5.1. The main idea of the PLSM model illustrated in Fig. 2 is that each occurrence of a scene activity leaves a noisy and variable trace in the word×time count matrix. The count matrix in Fig. 2(b) shows a simple case with observations from multiple occurrences of the two activities, making clear that activities can overlap in time, share the same vocabulary, occur without any particular synchronization, and are often accompanied with noise. Our goal in this difficult scenario is thus to recover the latent structure by learning the activity temporal patterns called motifs and their time of occurrence as illustrated in Fig. 2(b).



(a) Words from two activities shown as colored blobs.

(b) Temporal document d as a word×time count matrix $n(w, t_a, d)$, and on the right, the result of the PLSM algorithm (discovered activity motifs and their starting time).

Fig. 2: Applying PLSM to discover activities from videos. (a) Assume that the scene contains only two activities (Activity A - black and Activity B -red), and that we are able to automatically extract seven words labeled $\{w_1, \ldots, w_7\}$ depicted with colored blobs in the image. (b) Each activity occurrence leaves a (noisy) observation trail in a word×time count matrix according to a specific temporal pattern. For instance, in a simple case, activity A could be specified by the particular sequence of word $[w_6, w_3, w_4, w_5]$. Note that these trails can share vocabularies and can be interleaved, *i.e.,* have temporal overlap. The goal of the algorithm is thus, to identify the latent structure characterized by the activity motifs and their start times from the observed count matrix $n(w, t_a, d)$.

Symbol	Description
$\mathcal{D}$	Dataset, count matrices of the form $n(w, t_a, d)$
z	Motif index
w	Word index (SLA patterns for real data. See section 5.)
d	Temporal document index
t_a	Absolute time index in temporal documents
t_s	Start time of a motif
t_r	Relative time from the start of the motif
Θ	Model parameters $\{P(z\|d), P(t_s\|z, d), P(w, t_r\|z)\}$
N_z	Number of motifs
N_w	Vocabulary size (number of different words)
D	Number of temporal documents
T_z	Maximum duration of a motif
T_d	Duration of the temporal document
T_{ds}	Number of motif start time indices in a temporal document
$\lambda_{z,d}$	Sparsity constraint weight
λ_{bic}	Penalty term weight in BIC equation

Table 1: Notations used in this paper

3.2 Notation and model overview

Fig. 3(a) illustrates more formally the process to generate the temporal document matrix $n(w, t_a, d)$ that is qualitatively described in Fig. 2(b). The notations we follow in this paper is presented in Table 1.

Let D be the number of temporal documents in the corpus, with temporal documents indexed by d. Let $V = \{w_i\}_{i=1}^{N_w}$ be the vocabulary of words that can

occur at any given instant $t_a \in [1, \ldots, T_d]$, where N_w is the size of our vocabulary and T_d is the number of discrete time steps of temporal document d (and thus represents the duration of the temporal document). A temporal document is then described by its count matrix $n(w, t_a, d)$ indicating the number of times a word w occurs at the absolute time t_a within the temporal document d; a temporal document d thus contains $N_d = \sum_{w,t_a} n(w, t_a, d)$ words in total. According to our model, these temporal documents are generated from a set of N_z (N_z is the number of motifs) motifs $\{z_i\}_{i=1}^{N_z}$ represented by temporal patterns $P(w, t_r|z)$ with a fixed maximal duration of T_z time steps (*i.e.,* $t_r \in [0, \ldots, T_z - 1]$), where t_r denotes the relative time at which a word occurs within a motif. A motif can occur and start at any time instant $t_s \in [1, \ldots, T_{ds}]$ within the temporal document[1]. In other words, qualitatively, temporal documents are generated by taking the motifs and reproducing them in a probabilistic way (through sampling) at their starting positions within the temporal document, as illustrated in Fig. 2(b) and Fig. 3(a).

3.3 Generative Process

Our data $\mathcal{D}$ is the matrix $n(w, t_a, d)$ containing counts of triplets of the form (w, t_a, d). The actual process to generate these triplets (w, t_a, d) is given by the graphical model depicted in Fig. 3(b) and works as follows:

- draw a temporal document d with probability $P(d)$;
- draw a latent motif $z \sim P(z|d)$, where $P(z|d)$ denotes the probability that a word in temporal document d originates from motif z;
- draw the starting time $t_s \sim P(t_s|z, d)$, where $P(t_s|z, d)$ denotes the probability that the motif z starts at time t_s within the temporal document d;
- draw a word and relative time pair $(w, t_r) \sim P(w, t_r|z)$, where $P(w, t_r|z)$ denotes the joint probability that a word w occurs at time t_r within the motif z. Note that since $P(w, t_r|z) = P(t_r|z)P(w|t_r, z)$, this draw can also be done by first sampling the relative time from $P(t_r|z)$ and then the word from $P(w|t_r, z)$, as implied by the graphical model of Fig. 3(b);
- set $t_a = t_s + t_r$, which assumes that $P(t_a|t_s, t_r) = \delta(t_a - (t_s + t_r))$, that is, the probability density function $P(t_a|t_s, t_r)$ is a Dirac function. Alternatively, we could have modeled $P(t_a|t_s, t_r)$ as a noise process specifying uncertainty on the time occurrence of the word.

The main assumption with the above model is that, given the motifs, the occurrence of words within the document is independent of the motif start; that is, the occurrence of a word only depends on the motif, not on the time when a motif starts.

Before going into more details of the model, we present the terms that will be used in rest of this paper and establish the connection between the proposed model and its application to video activity analysis. Our input to the model is the pre-processed video represented by the word count matrix $n(w, t_a, d)$ in Fig. 2(b)

[1] The starting time t_s can range over different intervals, depending on hypotheses. In the experiments, we assumed that all words generated by a motif starting at time t_s occur within a temporal document; hence t_s takes values between 1 and T_{ds}, where $T_{ds} = T_d - T_z + 1$. However, we can also assume that motifs are partially observed (beginning or end are missing). In this case t_s ranges between $2 - T_z$ and T_d.

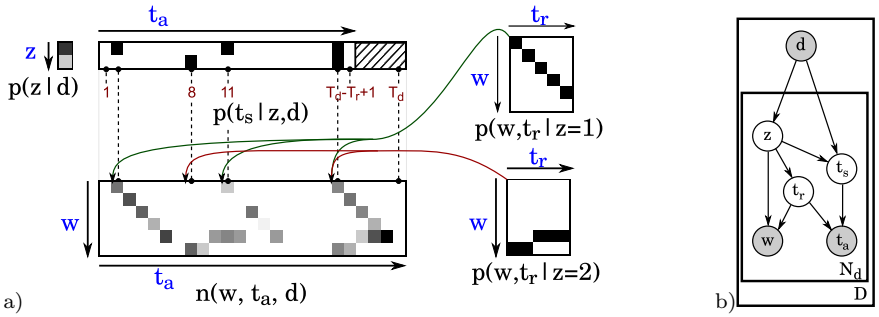


Fig. 3: Generative process. a) Illustration of the temporal document $n(w, t_a, d)$ generation. Words $(w, t_a = t_s + t_r)$ are obtained by first sampling the motifs and their starting times from the $P(z|d)$ and $P(t_s|z, d)$ distributions, and then sampling the word and its temporal occurrence within the motif from $P(w, t_r|z)$. b) Graphical model (shaded circles represent observed variables and unshaded ones indicate latent variables).

or 3(a), which we call a *temporal document* or simply a document. The motifs $P(w, t_r|z)$ used in the generative process are the dominant activities that occur in the scene as shown in Fig. 2(a) or repeated as in Figs. 15, 16 or 17. We refer to this distribution $P(w, t_r|z)$ as *motifs* due to the temporal aspect associated to each word and to distinguish them from simple word distributions $P(w|z)$ which are used in standard topic models like PLSA/LDA. Terms such as *dominant activities*, *activities*, *sequential patterns* are however, often exchanged with the term motifs. The words that appear in the temporal document and in the motifs could be application dependent. In case of our video surveillance application, we will consistently refer to these words as *Spatially Localized Activity* (SLA) patterns (see for instance the blobs shown in Fig. 2(a) or Fig 13). The method used to derive these SLA patterns is detailed in Section 5.1, and they themselves are obtained from quantized low-level visual features otherwise called low-level words. The generative process also infers when a motif occurs in the video using the motif start time distribution $P(t_s|z, d)$. This is indicated using the red and black arrows in Fig. 2(b) and is used to detect events in Section 6.3.

The joint distribution of all variables can be derived from the graphical model. However, given the deterministic relation between the three time variables $(t_a = t_s + t_r)$, only two of them are actually needed to specify this distribution. For instance, we have

$$\begin{aligned} P(w, t_a, d, z, t_s, t_r) &= P(t_r|w, t_a, d, z, t_s)P(w, t_a, d, z, t_s) \\ &= \begin{cases} P(w, t_a, d, z, t_s) & \text{if } t_r = t_a - t_s \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{1}$$

In the following, we will mainly use t_s and t_a. Accordingly, the joint distribution is given by:

$$P(w, t_a, d, z, t_s) = P(d)P(z|d)P(t_s|z, d)P(w, t_a - t_s|z). \tag{2}$$

3.4 Model inference with sparsity constraints

Our data $\mathcal{D}$ is the set of temporal documents $n(w, t_a, d)$. The likelihood of observing this data is given by the equation:

$$P(\mathcal{D}) = \prod_{d=1}^{D} \prod_{t_a=1}^{T_d} \prod_{w=1}^{N_w} P(w, t_a, d)^{n(w,t_a,d)} \tag{3}$$

From these observations, our goal is to discover the motifs and their starting times. This is a difficult task since the motif occurrences in the temporal documents overlap temporally, as illustrated in Fig. 3(a). The estimation of the model parameters Θ, *i.e.,* the probability distributions[2], $P(z|d)$, $P(t_s|z, d)$, and $P(w, t_r|z)$ can be done by maximizing the log-likelihood $\mathcal{L}(\mathcal{D}|\Theta)$ of the observed data $\mathcal{D}$. This is obtained by taking log on both sides of Eq. 3 and by marginalizing over the hidden variables $Y = \{t_s, z\}$ (since $t_r = t_a - t_s$, as discussed at the end of the previous subsection):

$$\mathcal{L}(\mathcal{D}|\Theta) = \sum_{d=1}^{D} \sum_{w=1}^{N_w} \sum_{t_a=1}^{T_d} n(w, t_a, d) \log \sum_{z=1}^{N_z} \sum_{t_s=1}^{T_{ds}} P(w, t_a, d, z, t_s) \tag{4}$$

However, as motivated in the introduction, the estimated distributions may exhibit a non-sparse structure that is not desirable in practice. In our model this is the case of $P(t_s|z, d)$: one would expect this distribution to be peaky, exhibiting high values for only a limited number of time instants t_s. To encourage this, we propose to guide the learning process towards sparser distributions by using a penalized likelihood optimization criterion.

There are several candidates for this penalty term. For instance, we could use a direct approach by minimizing the L_0 or L_1 norm of $P(t_s|z, d)$ considered as a vector of parameters, or use an entropy-based penalty term that favors low entropy and hence, sparse-distributions. However, such methods either do not suit well our probabilistic modeling approach (for instance, the L_1 norm of $P(t_s|z, d)$ can not be optimized as it is always equal to 1), or do not lead to simple optimization schemes [2]. Thus, we preferred to achieve this indirectly by adding a regularization constraint to maximize the Kullback-Leibler (KL) divergence $D_{KL}(U||P(t_s|z, d))$ between the uniform distribution U (maximum entropy) and the distribution of interest. Interestingly, in the past regularization approaches using KL-divergence to the uniform distribution have been used in physics [2], and have been shown to have good properties for sparse approximation, such as differentiability and increased stability of the solution [7]. In our case, the formulation we exploit has also the advantage of leading to a simple modification of the EM inference scheme.

Though such an approach can be applied to any distribution of the model, we demonstrate this approach by applying it to $P(t_s|z, d)$. After development and removing the constant term, our constrained objective function is now given by:

$$\mathcal{L}_c(\mathcal{D}|\Theta) = \mathcal{L}(\mathcal{D}|\Theta) - \sum_{t_s,z,d} \frac{\lambda_{z,d}}{T_{ds}} \cdot \log(P(t_s|z, d)) \tag{5}$$

[2] Note that $P(d) \propto N_d$ and is thus not unknown.

E-step:

$$P(z, t_s|w, t_a, d) = \frac{P(w, t_a, d, z, t_s)}{P(w, t_a, d)} \text{ with } P(w, t_a, d) = \sum_{z=1}^{N_z} \sum_{t_s=1}^{T_{ds}} P(w, t_a, d, z, t_s) \tag{7}$$

M-step:

$$P(z|d) \propto \sum_{t_s=1}^{T_{ds}} \sum_{t_r=0}^{T_z-1} \sum_{w=1}^{N_w} n(w, t_s + t_r, d) P(z, t_s|w, t_s + t_r, d) \tag{8}$$

$$P(t_s|z, d) \propto \max\left(\varepsilon, \left(\sum_{w=1}^{N_w} \sum_{t_r=0}^{T_z-1} n(w, t_s + t_r, d) P(z, t_s|w, t_s + t_r, d)\right) - \frac{\lambda_{z,d}}{T_{ds}}\right) \tag{9}$$

$$P(w, t_r|z) \propto \sum_{d=1}^{D} \sum_{t_s=1}^{T_{ds}} n(w, t_s + t_r, d) P(z, t_s|w, t_s + t_r, d) \tag{10}$$

Fig. 4: The EM algorithm steps.

where $\lambda_{z,d}$ denotes a weighting coefficient balancing the contribution of the regularization compared to the data log-likelihood.

As is often the case with mixture models, Eq. (5) can not be solved directly due to the summation terms inside the logarithm. Thus, we employ an Expectation-Maximization (EM) approach and maximize the expectation of the (regularized) complete log-likelihood instead, which is given by:

$$E[\mathcal{L}] = \sum_{d=1}^{D} \sum_{w=1}^{N_w} \sum_{t_a=1}^{T_d} \sum_{z=1}^{N_z} \sum_{t_s=1}^{T_{ds}} n(w, t_a, d) P(z, t_s|w, t_a, d) \log P(w, t_a, d, z, t_s) - \sum_{t_s,z,d} \frac{\lambda_{z,d}}{T_{ds}} \cdot \log(P(t_s|z, d)) \tag{6}$$

The solution is obtained by iterating the Eqs. (7–10) (see Appendix section. A for more details on this derivation). In the Expectation step, the posterior distribution of the hidden variables is calculated as in Eq. (7) where the joint probability is given by Eq. (2). In the Maximization step (Eqs. 8 to 10), the model parameters are updated by maximizing Eq. (6) along with the constraint that each of the distributions sum to one.

In practice, the EM algorithm is initialized using random values for the motif distributions (see also next subsection) and stopped when the data log-likelihood increase is too small. A closer look at the equations shows that in the E-step, the responsibilities of the motif occurrences (z, t_s) in explaining the word pairs (w, t_a) are computed (where high responsibilities will be obtained for informative words, *i.e.,* words appearing in only one motif and at a specific relative time), whereas the M-step aggregates these responsibilities to infer the motifs and their occurrences. Seen the other way round, the posterior terms $P(z, t_s|w, t_a, d)$ can be interpreted as weights or votes given to motif occurrences which are accumulated in Eqs. 9–10 to identify the relevant motif occurrences. Furthermore, by associating occurring words (w, t_a) to motif occurrences (z, t_s), this posterior implicitly aligns all the words of a motif instance with its starting time, and as a consequence statistically

achieves a soft alignment of multiple occurrences of the same motif, even in the presence of temporal overlap with the same or other motifs.

When looking at Eq. (9), we see that the effect of the additional sparsity constraint is to set to a very small constant ε the probability of terms which are lower than $\lambda_{z,d}/T_{ds}$ (before normalization), thus increasing the sparsity as desired. To set sensible values for $\lambda_{z,d}$ we used the rule of thumb $\lambda_{z,d} = \lambda\frac{n_d}{N_z}$, where n_d denotes the total number of words in the temporal document, and λ the sparsity level. Note that when $\lambda = 1$, the correction term $\lambda_{z,d}/T_{ds}$ in Eq. (9) is equal to the average (over t_s) of the term on the right hand side of Eq. (9) involving sums.

Inference on unseen temporal documents. Once the motifs are learned, their time occurrences in any new document – represented by $P(z|d_{new})$ and $P(t_s|z, d_{new})$, can be inferred using the same EM algorithm, but keeping the motifs fixed and using only Eq. (8) and Eq. (9) in the M-step.

3.5 Maximum a-posterior Estimation (MAP)

In graphical models, Bayesian approaches are often preferred compared to maximum-likelihood (ML) ones, especially if there is knowledge about the model parameters. This is the case for methods like LDA that can improve over PLSA by using Dirichlet priors on the multinomial distributions. However, as it was shown in [13] and [9], LDA is equivalent to PLSA when priors are uninformative or uniform, which is a common situation in practice.

The MAP estimation of parameters Θ can be formulated as follows:

$$\Theta_{\text{MAP}} = \arg\max_{\Theta}(\log P(\Theta|\mathcal{D})) = \arg\max_{\Theta}(\log P(\mathcal{D}|\Theta) + \log P(\Theta)) \tag{11}$$

where $P(\mathcal{D}|\Theta)$ is the likelihood term given by Eq. (4), and $P(\Theta)$ is the prior density over the parameter set. In practice, it is well known that using priors that are conjugate to the likelihood simplifies the inference problem. Since our data likelihood is defined as a product of multinomial distributions, we employ Dirichlet distributions as priors. A k dimensional random variable θ is said to follow a Dirichlet distribution parameterized by α if:

$$P(\theta|\alpha) \propto \prod_{i=1}^{k} \theta_i^{\alpha_i - 1} \tag{12}$$

where, $0 \leq \theta_i \leq 1, \forall i$ and $\sum_i \theta_i = 1$. Note that $\frac{\alpha}{\|\alpha\|_1}$ represents the expected values of the parameter θ (where $\|\alpha\|_1$ is the L1 norm of α), and, when the Dirichlet is used as a prior over the parameters θ of a multinomial distribution, $\|\alpha\|$ denotes the strength of the prior, and can indeed be viewed as a count of virtual observations distributed according to $\frac{\alpha}{\|\alpha\|_1}$.

Application to the PLSM model. Our parameter set Θ comprises the multinomial parameters $P(w, t_r|z)$, $P(z|d)$, and $P(t_s|z, d)$. We don't have any a priori information about the motif occurrences $P(t_s|z, d)$ nor can we obtain an updated prior that is common to all the temporal documents in a general scenario. Moreover, for this term, we employ the sparsity constraint rather than a smoothing prior. Thus, we will use the MAP approach to set priors on the other

multinomial parameters. Replacing in Eq. (5) the log-likelihood by the parameter log-posterior probability, the criterion to optimize simply becomes $\mathcal{L}_m(\mathcal{D}|\Theta) = \mathcal{L}_c(\mathcal{D}|\Theta) + \log P(\Theta)$, with the last term given by:

$$P(\Theta) \propto \prod_{d,z} P(z|d)^{\alpha_{z,d}-1} \prod_{z,w,t_r} P(w,t_r|z)^{\alpha_{w,t_r,z}-1}, \tag{13}$$

where $\alpha_{z,d}$ and $\alpha_{w,t_r,z}$ denote the Dirichlet parameters governing the prior distributions of $P(z|d)$ and $P(w,t_r|z)$ respectively. As before, $\mathcal{L}_m$ can be conveniently optimized using an EM algorithm, which leads to the same update expression as in Fig. 4, except that Eq. (8) and Eq. (10) need to be modified to account for the prior.

$$P_{\text{MAP}}(z|d) \propto (\alpha_{z,d} - 1) + \sum_{t_s=1}^{T_{ds}} \sum_{t_r=0}^{T_z-1} \sum_{w=1}^{N_w} n(w, t_s + t_r, d) P(z, t_s|w, t_s + t_r, d) \tag{14}$$

$$P_{\text{MAP}}(w,t_r|z) \propto (\alpha_{w,t_r,z} - 1) + \sum_{d=1}^{D} \sum_{t_s=1}^{T_{ds}} n(w, t_s + t_r, d) P(z, t_s|w, t_s + t_r, d) \tag{15}$$

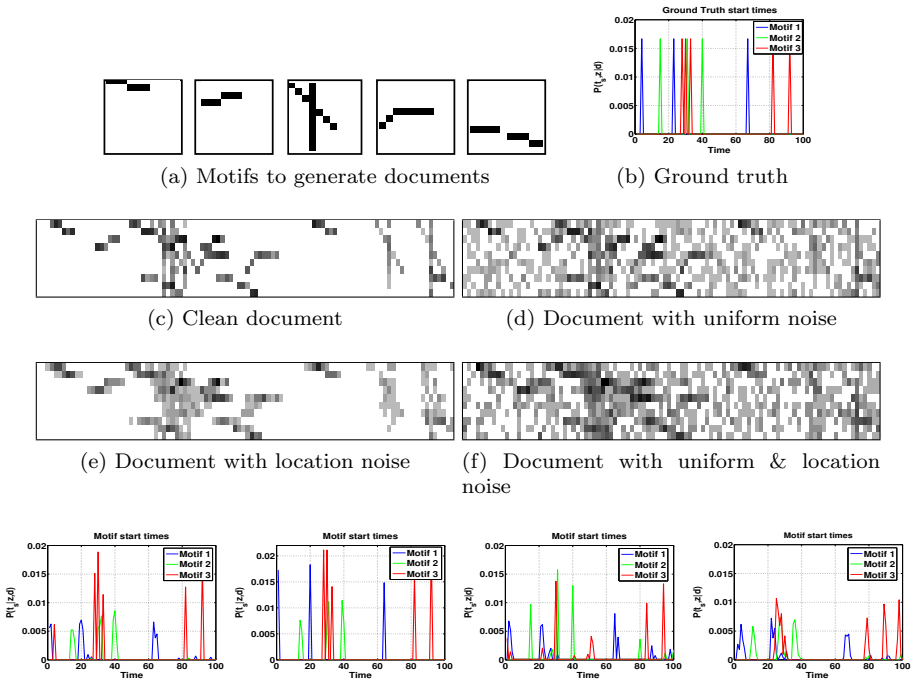
3.6 Model Selection

In unsupervised learning methods that are akin to clustering, the number of clusters is an important parameter to be determined. In our problem, this issue translates into identifying an appropriate number of motifs. Usually in real-life scenarios, we have some rough a-priori knowledge of the number of motifs. This is the case, for instance, in our video activity analysis scenarios, where this number qualitatively depends on the scene complexity, the types of features (observations), and the duration of the sought motifs. Still, being able to adapt the selected number of motifs as a function of the actual data is desirable.

There are several methods that can be used for model selection in unsupervised settings. They include testing on held-out data [3], the Bayesian Information Criterion (BIC) [31], and more sophisticated non-parametric approaches like Hierarchical Dirichlet Processes [35][10]. In this work, we use the BIC measure, which penalizes the training data likelihood based on the number of parameters and data points. A general version of the BIC measure of a model M is given by:

$$BIC(M) = -2\mathcal{L}(\mathcal{D}|\Theta) + \lambda_{bic} N_p^M \log(n) \tag{16}$$

where, $\mathcal{L}$ is the likelihood of the model and is given by Eq. (4), N_p^M denotes the number of parameters of model M, n is the number of data points, and λ_{bic} is a coefficient that controls the influence of the penalty. Note that the above equation leads to the standard BIC criterion when λ_{bic}, the weight of the penalty term, is 1. In practice, we followed the approach of [8,37] and used a validation set to set this parameter in the numerical experiment. In essence, this criterion seeks models that find a compromise between likelihood fitting and model complexity. In practice, we conduct optimization for models with different number of motifs according to previous subsections, and finally keep the model with the minimum BIC measure.



(a) Motifs to generate documents (b) Ground truth

(c) Clean document (d) Document with uniform noise

(e) Document with location noise (f) Document with uniform & location noise

(g) Clean, $\lambda = 0.0$ (h) Clean, $\lambda = 0.5$ (i) Uniform, $\lambda = 0.5$ (j) Location, $\lambda = 0.5$

Fig. 5: Synthetic experiments. (a) The five motifs, (b) the true motif occurrences (only 3 of them are shown for clarity) used to generate documents; (c) a segment of a generated document; (d,e) the same segment perturbed with: (d) Uniform noise ($\sigma_{\text{snr}} = 1$); (e) location noise ($\sigma = 1$) added to each word time occurrence t_a; (f) a document segment with both uniform ($\sigma_{\text{snr}} = 1$) and location noise ($\sigma = 1$); (g–j) the recovered motif occurrences $P(t_s|z,d)$: (g) from the clean document (cf c) with no sparsity constraint $\lambda = 0$; (h) from the clean document with sparsity constraint $\lambda = 0.5$; (i) from the noisy document (d) with $\lambda = 0.5$; (j) from the noisy document (e) with $\lambda = 0.5$.

4 Experiments on synthetic data

In order to investigate various aspects of the model and validate its strengths we first conducted experiments using synthetic data.

4.1 Data and experimental protocol

Data synthesis. Using a vocabulary of 10 words, we created five motifs with duration ranging between 6 and 10 time steps (see Fig. 5(a)). Then, for each experimental condition (*e.g.,* a noise type and noise level), we synthesized 10 documents of 2000 time steps following the generative process described in section 3.3, assuming equi-probable motifs and 60 random occurrences per motif. A one hundred time-step segment of a clean document is shown in Fig. 5(c), where the

intensities represents the word count (larger counts are darker). In Fig. 5(b) corresponding starting times of the first three motifs out of the five motifs are shown for the sake of clarity. Note that there is a large amount of overlap between motifs.

Adding noise. Two types of noise were used to test the robustness of the method. In the first case, words were added to the clean documents by randomly sampling the time instant t_a and the word w from a uniform distribution, as illustrated in Fig. 5(d). We call this *Uniform noise*. Here, the objective is to measure the algorithm performance when the ideal co-occurrences are disturbed by random word counts. The amount of noise is quantified by the ratio $\sigma_{\text{snr}} = N_w^{noise}/N_w^{true}$ where, N_w^{noise} denotes the number of noise words added and N_w^{true} is the number of words in the clean document. In practice, noise can also be due to variability in the temporal execution of the activity. Thus, in the second case, a *Location noise* was simulated by adding random shifts sampled from Gaussian distribution with $\sigma \in [0, 2]$ to the time occurrence t_a of each word, resulting in blurry documents, as shown in Fig. 5(e). As a third case, documents with both Uniform and Location noise were also created. To this end, documents with a fixed location noise $\sigma = \{0.5, 1.0\}$ were first created. Then, uniform noise of varying strengths $\sigma_{snr} = \{0.5, 1, 1.5, 2\}$ were added on these already noisy documents. One sample document is shown in Fig. 5(f). Notice how the motifs have become almost indistinguishable from noise.

Model parameterization. As we do not assume any prior on the parameter model, we did not use the MAP approach in these experiments, and optimized the penalized likelihood of Eq. (5). For each document, 10 different random initializations were tried and the model maximizing the objective criterion was kept as the result.

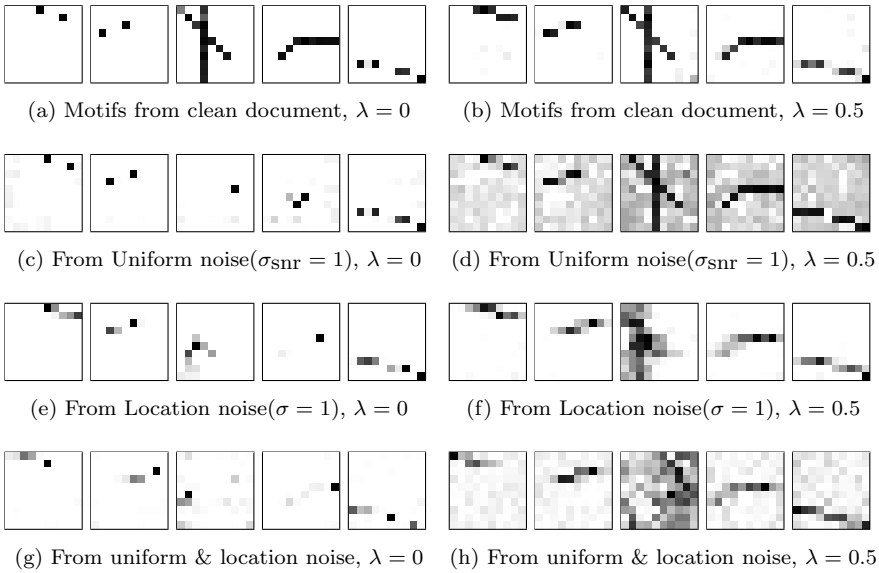
Performance measure. The learning performance is evaluated by measuring the normalized cross correlation.[3] Averages and corresponding error-bars computed from the results obtained on the 10 generated documents are reported.

4.2 Results

Results on clean data. Figs 6(a) and 6(b) illustrate the recovered motifs without and with the sparsity constraint respectively. As we can see, without sparsity, two of the obtained motifs are not well recovered. This can be explained as follows. Consider the first of the five motifs. Samples of this motif starting at a given instant t_s in the document can be equivalently obtained by sampling words from the learned motif in Fig. 6(a) and sampling the starting time from three consecutive t_s values with probabilities less than one. This can be visualized in Fig. 5(g), where the peaks in the blue curve $P(t_s|z = 1, d)$ are three times wider and lower than in the ground truth. When using the sparsity constraint, the motifs are well recovered, and the starting time occurrences better estimated, as seen in Fig. 5(h) and Fig. 6(b).

Robustness to noise. Fig. 6(c,e,g) illustrate the recovered motifs under noise, without sparsity constraint. We can clearly observe that the motifs are not well

[3] The correspondence between the ground truth motifs and the estimated ones is made by optimizing the normalized cross-correlation measure between the learned motifs $\hat{P}(t_r, w|z)$ and the true motifs $P(t_r, w|z)$.



(a) Motifs from clean document, $\lambda = 0$ (b) Motifs from clean document, $\lambda = 0.5$

(c) From Uniform noise($\sigma_{\text{snr}} = 1$), $\lambda = 0$ (d) From Uniform noise($\sigma_{\text{snr}} = 1$), $\lambda = 0.5$

(e) From Location noise($\sigma = 1$), $\lambda = 0$ (f) From Location noise($\sigma = 1$), $\lambda = 0.5$

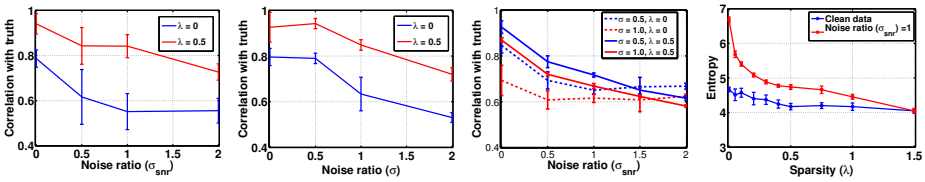
(g) From uniform & location noise, $\lambda = 0$ (h) From uniform & location noise, $\lambda = 0.5$

Fig. 6: Synthetic experiments. Recovered motifs without (a,c,e,g) and with (b,d,f,h) sparsity constraints, $\lambda = 0.5$, under different noise conditions; (a,b) from clean data; (c,d) from documents perturbed by Uniform noise, $\sigma_{\text{snr}} = 1$, cf Fig. 5(d); (e,f) from documents perturbed with Location noise $\sigma = 1$, cf Fig. 5(e); (g,h) motifs from documents with both uniform $\sigma_{\text{snr}=1}$ and location noise $\sigma = 1$, cf Fig. 5(f); (g) without sparsity; (h) with sparsity $\lambda = 0.5$.

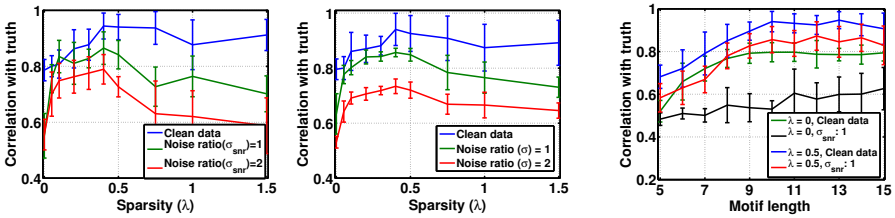
recovered (*e.g.,* the third motif is completely missed in all the three cases). With sparsity, Fig. 6(d,f,h) motifs are better recovered, but reflect the presence of the generated noise, *i.e.,* the addition of uniform noise in the motifs in the first case, the temporal blurring of the motifs in the second case and a combination of both noises in the third case respectively. The curves in Fig. 7(a,b,c) show the degradation of the learning as a function of the noise level.

Effect of sparsity. We also analyzed the performance of the model by varying the weight of the sparsity constraint for different noise levels and noise types. Fig. 7(a,b,c) show that the model is able to handle quite a large amount of noise in all the three cases, and that the sparsity approach always provides better results. In Fig. 7(a) while the best results without the constraint gives only a correlation of 0.8, we achieve a much better performance (approximately 0.95) with sparsity. In the very challenging case where both uniform and location noise are simultaneously present, cf. Fig. 6(h) and Fig. 7(c), we see that four of the five motifs have been recovered quite well even under such noisy conditions with a correlation of nearly 0.65 when $\sigma = 1$ and $\sigma_{snr} = 1$.

In Fig. 7(e) and 7(f), we see the performance of the method for various values of the sparsity weight λ and for varying noise levels. We notice that as the weight for sparsity increases, the performance shoots up. However, an increase of the sparsity weight beyond 0.5 often leads to degraded and sometimes unstable performance.



(a) Uniform Noise (b) Location Noise (c) Uniform & Location noise, $\lambda = 0.5$ (d) Average entropy



(e) Uniform Noise (f) Location Noise (g) Changing motif length

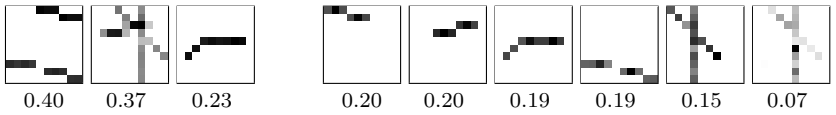
Fig. 7: Effect of sparsity constraint: (a,b,c) Average motif correlation between the estimated and the ground truth motifs for different levels of (a) Uniform noise; (b) Location noise; (c) Uniform and location noise; (d) Average entropy of $P(t_s|z,d)$ as a function of sparsity weight λ; (e,f) Average motif correlation between the estimated and the ground truth motifs; for different sparsity weight λ and for different levels of (e) Uniform noise; (f) Location noise on a word time occurrence t_a; (g) Effect of varying motif length T_z from 5 to 15, for two levels of uniform noise.

Finally also note, as illustrated by Fig. 7(d), that the increase of the sparsity weight λ leads to a lowering of the entropy of $P(t_s|z,d)$, as desired.

We conclude from these results that we obtain a marked improvement in recovering the motifs from both clean and noisy documents when sparsity constraint is used.

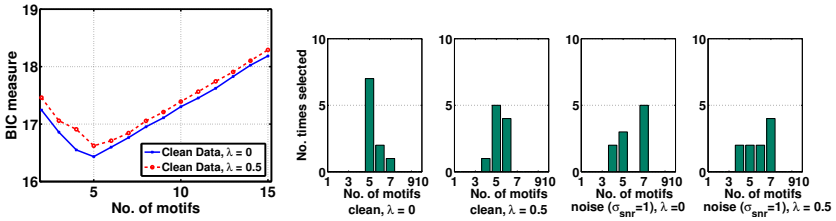
Number of motifs and model selection. We first studied the qualitative effect of changing N_z, the number of motifs. As illustrated in Fig. 8. When N_z is lower than the true number, we observe that each estimated motif consistently captures several true motifs. For instance, the first motif in Fig. 8(a) merges the 1st and 5th motif of Fig. 5(a). When the number of motifs is larger than the true value, like $N_z = 6$ in the example, we see that a variant of one motif is captured, but with lower probability. We observe the same phenomenon as we further increase the number of motifs.

We also tested our model selection approach based on the BIC criteria, as explained in section 3.6. To set λ_{bic}, we generated five extra clean documents and used them to select an appropriate value of this parameter. Then, the same value was used to perform tests on other clean or noisy documents. Fig. 9(a) displays the BIC values obtained for a clean document by varying the number of motifs from 2 to 15. As we see, the criteria reaches its minimum for 5 motifs. Histograms in Fig. 9(b) show the number times a motif size is selected for a set of documents. Although not perfect, the results show that the method is able to retrieve an



(a) Three motifs (b) Six motifs

Fig. 8: Effect of changing N_z. Estimated motifs sorted by their $P(z|d)$ values (given below each motif) when the number of motifs is (a) $N_z = 3$ - true motifs are merged; (b) $N_z = 6$ - a duplicate version of a motif with slight variation is estimated.



(a) BIC measure (b) Histogram of number of motifs selected

Fig. 9: Bayesian Information Criteia. (a) Example of a BIC measure on a synthetic document. (b) Selection histograms (number of times a motif is selected) using BIC on 10 documents, with the following conditions: clean, $\lambda = 0$, clean, $\lambda = 0.5$, Noise ($\sigma_{\text{snr}} = 1$), $\lambda = 0$ and Uniform noise ($\sigma_{\text{snr}} = 1$), $\lambda = 0.5$.

appropriate number of motifs. In the presence of strong noise with as many noise words as true words, the number of found motifs is usually larger. This is expected as we need more motifs to explain the additional noise in the data.

Motif length. The effect of varying the maximum duration T_z of a motif and in the presence of noise is summarized in Fig. 7(g). When T_z becomes lower than the actual motif duration, the recovered motifs are truncated versions of the original ones, and the "missing" parts are captured elsewhere, resulting in a decrease in correlation. On the other hand, longer temporal windows do not really affect the learning, even under noisy conditions. However, the performance under clean and noisy conditions are significantly worse with no sparsity constraint. From this experiment we can infer that setting a longer motif duration could be a better choice, especially when we do not have an idea of the true motif duration.

Comparison with TOS-LDA [24]. TOS-LDA works by collecting independent Bag-of-Words (BoW) documents from the video and then by applying an LDA topic discovery method to these documents. Figure 10 illustrates how the BoW documents of the TOS-LDA method are constructed from the temporal documents shown in Fig. 2(b). These documents are simply obtained from windows of a fixed temporal duration swept over the video, and by associating to each word w_i a time stamp t^r_j relative to the start of each fixed-size window. In other words, the TOS vocabulary is defined as the set of words $w^{\text{TOS}}_{ij} = (w_i \times t^r_j)$. In Fig. 10, we show three sample documents created from the video by sliding a window of 4 time steps duration over three time steps. The first document is made of the TOS words $\{w^{\text{TOS}}_{61}, w^{\text{TOS}}_{32}, w^{\text{TOS}}_{43}, w^{\text{TOS}}_{54}\} = \{(w_6, t^r_1), (w_3, t^r_2), (w_4, t^r_3), (w_5, t^r_4)\}$, and

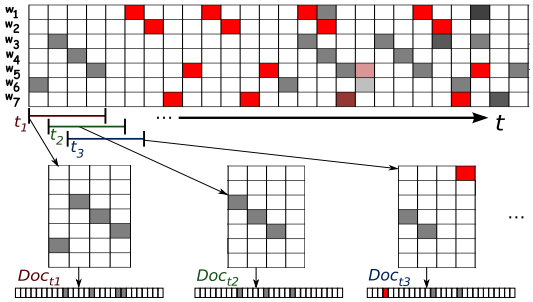


Fig. 10: Illustration of the method [24]. Individual TOS Bag-of-Words (BoW) documents $Doc_{t_1}, Doc_{t_2}, Doc_{t_3}, ...$ are created from the video count matrix by sliding a window over time, and considering each pair (word×relative-time-to-the-window-start) in these windows as a TOS word w_{ij}^{TOS}. In the example, the TOS vocabulary is thus of size $7 \times 4 = 28$ (*i.e.,* documents are defined as counts of these w_{ij}^{TOS} words). Note how with TOS-LDA, the same motif occurrence in the video can result in different set of BoW representation depending on when the motif started with respect to the start of the document window, and that all documents are considered to be independent.

Fig. 11: Five topics obtained from the TOS-LDA method [24] with clean data.

similarly, the second document contains $\{w_{31}^{\text{TOS}}, w_{42}^{\text{TOS}}, w_{53}^{\text{TOS}}\}$, and so on for other documents. Thus, in this approach, one can clearly see that the same observed activity results in different sets of words for each document depending on its relative time occurrence within these sliding windows (in the example, documents 1 and 2 have orthogonal representations although they contains the same sub-activity from the video). In other words, in the learning, *several* motifs (being time shifted versions of each other) will be needed to capture the *same* activity and account for the different times at which it can occur within the window. As pointed out in the related work section 2, the method clearly lacks an alignment procedure that indicates when the motif starts, an information that is manually supplied in [12] for activities based on traffic cycles.

We applied TOS-LDA to the same set of synthetic documents created as previously described, and Fig. 11 shows the obtained motifs when using clean data. Due to the method's inherent lack of alignment ability, none of the five extracted TOS-LDA topics truly represents one of the five motifs used to create the documents. Rather, they contain parts, blurry and mixed versions of them, with some of the motifs (*e.g.,* the vertical bar) not appearing at all.

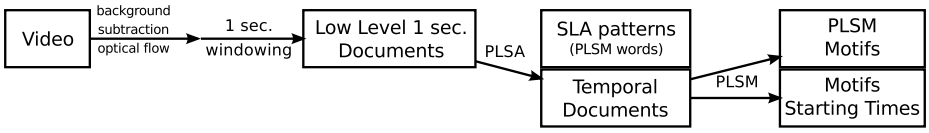


Fig. 12: Flowchart for discovering sequential activity motifs in videos. Quantized low-level features are used to build 1 second bag-of-words documents, from which Spatially Localized Activity patterns (SLA) are learned and further used to build the documents used as input to PLSM.

5 Application to video scene activity analysis

Our objective is to identify recurring activities in video scenes from long term data automatically. In this section, we explain how we can use the PLSM model for this purpose, and describe the video preprocessing used to define the words and documents required by the PLSM model. We then present the datasets used for experiments and finally show three different ways of representing the learned motifs.

5.1 Activity word and document construction

To apply the PLSM model to videos, we need to specify its inputs: the words w forming its vocabulary and that define the semantic space of the learned motifs, and the corresponding documents. One possibility would be to define quantized low-level motion features and use these as our words. However, this would result in a redundant and unnecessarily large vocabulary. As an alternative, we propose to first perform a dimensionality reduction step by extracting spatially localized activity (SLA) patterns from the low-level features and use the occurrences of these as our words to discover activity motifs using the PLSM model. To do so, we use the approach in [46, 41] and apply a standard PLSA procedure to discover N_A dominant SLA patterns that consists of co-ocurring low-level visual words w^{ll}. The work flow of this process is shown in Fig. 12. Note that in addition to dimensionality reduction, the PLSA approach, being data driven, will also create a vocabulary that is well adapted to the actual scene content, as will be illustrated below.

Low-level words w^{ll}. The low-level words come from the location cue (quantized into 2×2 non-overlapping cells) and motion cue. First, to identify foreground pixels, background subtraction is performed using the adaptive multi-layer background subtraction method by [53]. For the foreground pixels, optical flow features are computed using the Lucas-Kanade algorithm [36]. The foreground pixels are then categorized into either static pixels (static label) or pixels moving into one of the eight cardinal directions by thresholding the flow vectors. Thus, each low-level word $w^{ll}_{c,m}$ is implicitly indexed by its location c and motion label m. Note that the static label will be extremely useful for capturing waiting activities, which contrasts with previous works [46].

SLA patterns z^{ll}. We apply the PLSA algorithm on a document-word frequency matrix $n(d_{t_a}, w^{ll})$ obtained by counting for the document d_{t_a} the low-level words

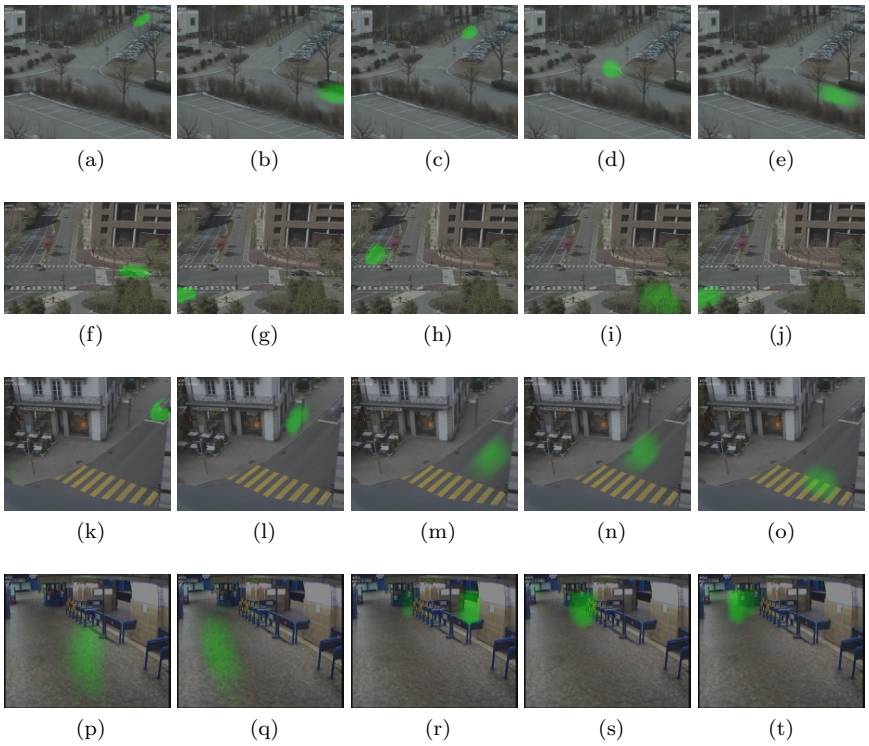


Fig. 13: Representative SLA patterns obtained by applying PLSA on (a–e) far-field data, (f–j) MIT data and (k–o) Traffic junction data, (p–t) Metro station data.

appearing in N_f frames within a time interval of one second centered on time t_a. The result is a set of N_A SLA patterns characterized by their multinomial distributions $P(w^{ll}|z^{ll})$, and the probabilities $P(z^{ll}|d_{t_a})$ providing the topic distribution for each document. While PLSA captures dominant low-level word patterns, it can also be viewed as a data reduction process since it provides a much more concise way of representing the underlying activities in the video at a given instant t_a, using only N_A SLA patterns, a number much smaller than the low-level vocabulary size. We observed that, with 50 to 100 SLA patterns, we get an accurate description of the scene content. We also ran HDP [35], which automatically finds the number of topics, and we obtained between 20 and 60 SLA patterns depending on the parameters and the dataset. Following these observations, we used $N_A = 75$ to get both a good representation of the scenes and a reasonable complexity for PLSM processing.

We can visualize the result of this step by superimposing the distributions $P(w^{ll}|z^{ll})$ over the image, indicating the locations where they have high probabilities. This is illustrated in Fig. 13, which shows representative SLA patterns obtained from each of the four video scenes described below, with their locations

highlighted in green[4]. Clearly, the SLA patterns represent spatially localized activities in the scene.

Building PLSM documents. In our approach, we define the PLSM words as being the SLA patterns (*i.e.,* we have $w \leftrightarrow z^{ll}$ and $N_w = N_A$). Thus, to build the documents d for PLSM, we need to define our word count matrix $n(w, t_a, d)$ characterizing the amount of presence of the SLA patterns z^{ll} in the associated low-level document at this instant t_a, *i.e.,* d_{t_a}. To do so, we exploit two types of information: the overall amount of activity in the scene at time t_a, and how this activity is distributed amongst the SLA patterns. The word counts were therefore simply defined as:

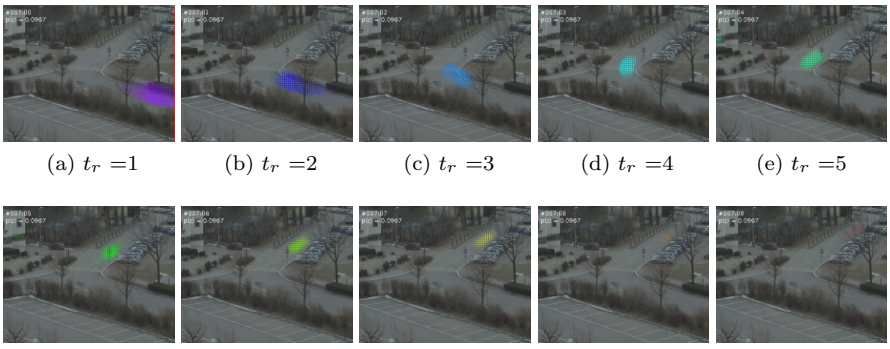
$$n(d, t_a, w) = n(d_{t_a}) P(z^{ll} | d_{t_a}) \tag{17}$$

where $n(d_{t_a})$ denotes the number of low-level words observed at a given time instant (*i.e.,* within the 1 second interval used to build the d_{t_a} document).

5.2 Video datasets

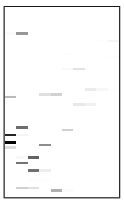
Experiments were carried out on four complex scenes with different activity contents obtained from fixed cameras. The **MIT** scene [46] is a two-lane, four-road junction captured from a distance, where there are complex interactions among vehicles arriving from different directions, and few pedestrians crossing the road (see Fig. 1(a)). This has a duration of 90 minutes, recorded at 30 frames-per-second (fps), and a resolution of 480×756 which was down-sampled to half its size. The **Far-field** scene [38] depicts a three-road junction captured from a distance, where typical activities are moving vehicles (see Fig. 1(b)). As the scene is not controlled by a traffic signal, activities occur at random. The video duration is 108 minutes, recorded at 25 fps and a 280×360 frame resolution. The **Traffic junction** [41] (see Fig. 1(c)) captures a portion of a busy traffic-light-controlled road junction. In addition to vehicles moving in and out of the scene, activities in this scene also include people walking on the pavement or waiting before walking across the pedestrian crossing. The video, recorded at 25 fps and a 280×360 frame resolution, has a duration of 44 minutes. The **Metro station** data (Fig. 1(d)) is captured from a static camera looking at a hallway, with people arriving there from several directions, buying tickets, staying in the hall, or going through turnstiles leading to the train platform. The scene is usually crowded with a high degree of unstructured movement by people. More importantly, due to the low view point, motion at a given image location can be due to people moving at different depths in the scene, making the low level image measurements highly ambiguous. The video is 120 minutes long and captured at 5 fps with a frame resolution of 576×720. Due to a high degree of noise in this scene, our features are only from optical-flow without considering back ground subtraction.

[4] Note that the topic distributions contain more information than the location probability: for each location, we know what types of motion are present as well. This explains the location overlap between several topics, *e.g.,* between those of Fig. 13(b) and Fig. 13(e), which have different dominant motion directions.

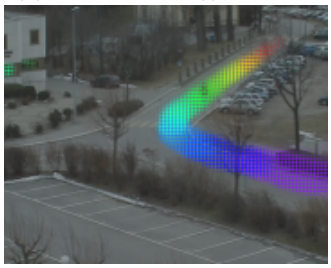


(a) t_r =1 (b) t_r =2 (c) t_r =3 (d) t_r =4 (e) t_r =5

(f) t_r =6 (g) t_r =7 (h) t_r =8 (i) t_r =9 (j) t_r =10



(k) $P(w, t_r|z)$



(l) collapsed t_r =1:10

Fig. 14: Three different representations of a PLSM motif. (k) Motif probability matrix. The x axis denotes t_r, and the y axis the words. (a-j) For each time step t_r, weighted overlay on the scene image of the locations associated to each word (*i.e.,* the SLA patterns). (l) All time steps collapsed into one image color-coded according to the rainbow scheme, (Violet for $t_r = 1$ to Red for $t_r = T_z$).

5.3 Motif representation

Before looking at the results obtained from the datasets, we explain how the learned motifs are represented visually. In Fig. 14, we provide three different ways of representing a recovered motif of $T_z = 10$ time steps (seconds) duration obtained from PLSM. By definition, a PLSM motif is a distribution $P(w, t_r|z)$ over $w \times t_r$ space. Thus the direct depiction of the motif is that of the $P(w, t_r|z)$ matrix as given in Fig. 14(k). This shows that the distribution is relatively sparse, words often occur at several consecutive time steps, and that several words co-occur at each time step. However, this does not provide much intuition about the activities captured by the motif. The second way of representing the motif is to back-project on the scene image and for each time step t_r, the locations associated with the words (the SLA patterns) probable at this time step, similar to the illustration of the SLA patterns in Fig. 13. This is illustrated in Fig. 14(a–j). This provides a good representation of the motif, but is space consuming. An even more realistic representation giving a true grasp of the motifs is provided by rendering them as animated gifs. This is what we provide in the additional material that is hosted

in a permanent institution web-page on `http://www.idiap.ch/paper/plsm/plsm.html`.

Due to media and space limitations, we use here an alternative version of these representations that collapses all time steps into a single image using a color-coded scheme, as shown in Fig. 14(l). Note that the color at a given location is the one of the largest time step t_r for which the location probability is non zero. Hence, the representation may hide some local activities due to the collapsing effect. However, in the large majority of cases, the representation provides good intuition of the learned activities.

6 Video Scene Analysis Results

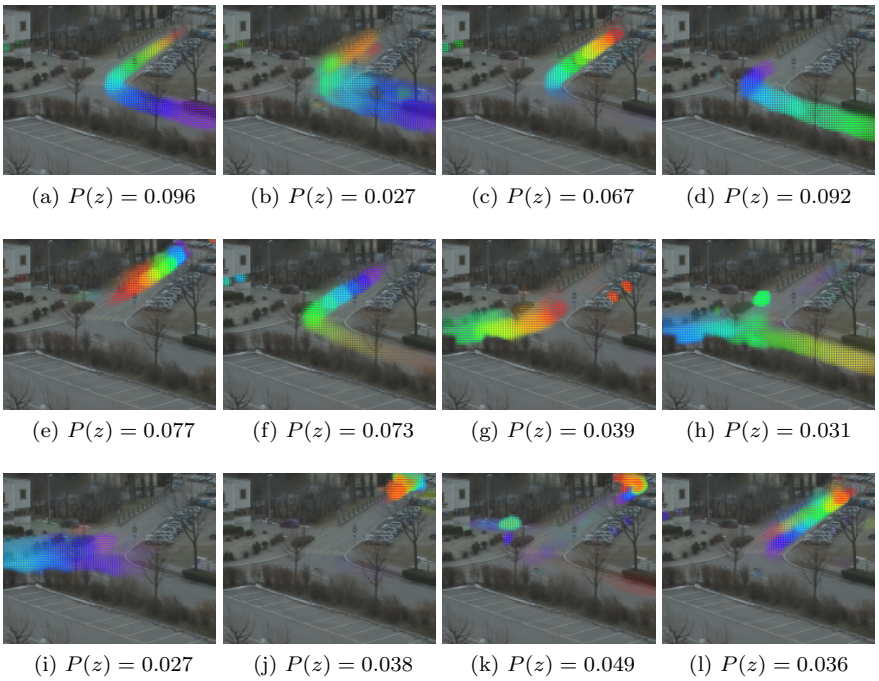
In this section, complementary details about the algorithm implementation are provided. Then, recovered motifs on the four datasets are shown and commented on. We then report the results of quantitative experiments on a counting task and on a prediction task to further validate our approach.

6.1 Experimental details

Since our method is unsupervised, we used all the data to learn the SLA patterns and PLSM parameters. To learn the SLA patterns we used documents created from video clips of 1 second duration. To reduce the computational cost, optical flow features were estimated and collected in only $N_f = 5$ frames of these intervals. As for the number of PLSM temporal documents and their length T_d, this does not affect our modeling. Therefore, one may choose to use the entire video as a single temporal document. However, in practice, we used temporal documents created from 2 minutes of video clips ($T_d = 120$) for the urban datasets, and from 10 minute video clips ($T_d = 600$) for the metro dataset. A notable exception to using all the data is when we evaluate the model in section 6.3. Here, we use only parameters trained from 90% of the data and test it on the remaining 10% of the data. This is repeated for all the 10 folds.

To favor the occurrence of the word probability mass at the start of the estimated motifs, we relied on the MAP framework and defined Dirichlet prior parameters for the motifs[5] as $\alpha_{w,t_r,z} = \tau \cdot \frac{1}{N_z} \cdot f(t_r)$, where f denotes a normalized (*i.e.,* the values of $f(t_r)$ sums to 1) decreasing ramp function as $f(t_r) \propto (T_z - t_r) + c$, T_z is the motif duration and c is a constant term. In other words, we did not impose any prior on the word occurrence probability, only on the time when they can occur. The strength of the prior is given by the term τ and was defined as a small fraction (we used 0.1) of the average number of observations in the training data for each of the $N_z \cdot N_w \cdot T_z$ motif bins. In practice, the prior plays a role when randomly drawing the motifs at initialization, where they are generated from the prior, and during the first few EM iterations. After, given the (low) level of the τ value and the concentration of the real observations on a few motif bins (see an estimated motif in Fig. 14), its influence becomes negligible. More details on the effect of this MAP prior is in the later part of this section and illustrated in Fig. 20.

[5] Note that we did not set any prior on the motif occurrences within the document, *i.e.,* we set $\alpha_{z,d} = 0$.



(a) $P(z) = 0.096$ (b) $P(z) = 0.027$ (c) $P(z) = 0.067$ (d) $P(z) = 0.092$

(e) $P(z) = 0.077$ (f) $P(z) = 0.073$ (g) $P(z) = 0.039$ (h) $P(z) = 0.031$

(i) $P(z) = 0.027$ (j) $P(z) = 0.038$ (k) $P(z) = 0.049$ (l) $P(z) = 0.036$

Fig. 15: Far-field data. Twelve representative motifs of 10s duration, out of 20. The method is able to capture the different vehicular trajectory segments. Best viewed in color. Please see the additional material [1] to view animated gif versions of the motifs.

With regards to sparsity weight, since there is a large range of values from 0.15 to 0.5 that leads to good results, we set λ to 0.5 in all our real data experiments.

6.2 PLSM motifs and activities

We first sought for motifs of maximum 10 seconds duration, *i.e.*, $T_z = 10$. Note that 10 seconds already capture relatively long activities, especially when dealing with vehicles. At the end of this section, we also show results for 20 second motifs.

The number of 10s motifs selected automatically using the BIC criteria were 20, 26, 16 and 25 for the Far-field, MIT, Traffic junction and Metro Station datasets respectively. A selection of the top-ranking representative motifs are shown in Fig. 15, Fig. 16, Fig. 17 and Fig. 18 using the collapsed color representation (cf Fig. 14), along with their probability $P(z)$ in explaining the training data.[6] Below we comment on the results.

Far-field data. The analysis of the motifs show the ability of the method to capture the dominant vehicle activities and their variations due to differences of

[6] In the the additional material an exhaustive set of results are provided with motifs rendered in animated-GIF.

trajectory, duration, and vehicle type, despite the presence of trees at several places that perturb the estimation of the optical flow. For instance, Fig. 15(a–c) correspond to vehicles moving towards the top right of the image, and Fig. 15(d–f) corresponds to vehicles moving from the top right. Fig. 15(g) corresponds to vehicles moving from left of the scene to the top right, Fig. 15(h) shows vehicles moving from left to bottom and Fig. 15(i) shows movement towards the left.

Some of the motifs capture the full presence of a vehicle in the scene (e.g, Fig. 15(h)) but most of the activities are longer than the motif duration (10 seconds). The only solution for the algorithm is thus to split the activities in multiple motifs. For example motifs Fig. 15(g,c and k) together cover the complete trajectory of a car going from the left to the top right of the scene. We observe that the algorithm tend to factor out the common subparts of the trajectories, for example motif Fig. 15(c) is also used for cars coming from the bottom right and going to the top right. The split of trajectories becomes unnecessary when we increase the motif duration, *e.g.,* to 20 seconds as in Fig. 19.

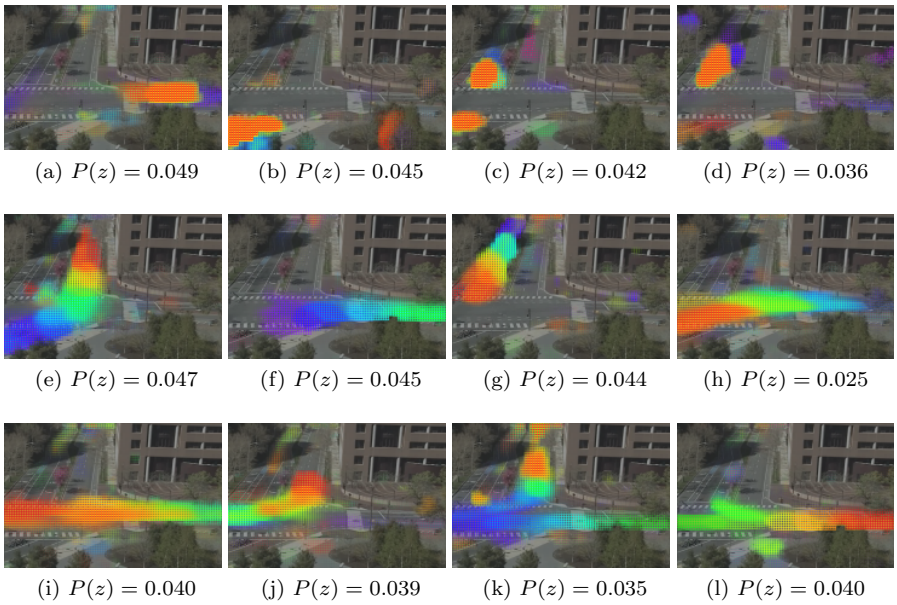
We also see that vehicle speed has an impact on the recovered motifs. When the possible speed differences are important, multiple motifs are recovered for the different speeds. In Fig. 15, this is the case for Fig. 15(a) and Fig. 15(b) which differ in the distance crossed by the motifs and also in the size of the vehicle. More cases of speed variation can be found in additional material [1].

Motifs in Fig. 15(j,k) represent the activities of vehicles moving in and out of the scene at the top of the scene. Since this location is far from the camera, and vehicles in both directions have to slow down due to a bump in the road, their apparent motion in the image is very slow and all the words are concentrated over a small region for the entire motif duration. Finally, the motif in Fig. 15(l) represents the activity of two vehicles passing each other on the top part of the road.

MIT data. This dataset is quite complex, with multifarious activities occurring concurrently and being only partially constrained by the traffic light. Even in this case, our method extracted meaningful activities corresponding to the different phases of the traffic signal cycle, as shown in Fig. 16. Briefly speaking, one finds two main activity types: waiting activities, shown in Fig. 16(a-d)[7], and dynamic activities as shown in Fig. 16(e-l) of vehicles moving from one side of the junction to the other after the lights change to green. Note that waiting activities that were not captured in previous works like [46], are identified here thanks to the use of background subtraction and of static words.

Traffic junction data. Despite the small amount of data (44min) and complex interactions between the objects of the scene, the method is able to discover the dominant activities as shown in Fig. 17. These are for instance dynamical activities due to vehicles, which usually last around 5 seconds only which explains the absence of the whole color range in Fig. 17(a-c). Note that while Fig. 17(a) corresponds to cars going straight, Fig. 17(b) shows cars coming from the top right and turning to their right at the bottom. Waiting activities are also captured, as illustrated in the motif of Fig. 17(d), which displays vehicles waiting for the

[7] Waiting activities are characterized by the same word(s) repeated over time in the motif. Thus the successive time color-coded images overwrite the previous ones in the collapsed representation as explained in Section. 5.3, leaving visible only the last (orange, red) time instant.



(a) $P(z) = 0.049$ (b) $P(z) = 0.045$ (c) $P(z) = 0.042$ (d) $P(z) = 0.036$

(e) $P(z) = 0.047$ (f) $P(z) = 0.045$ (g) $P(z) = 0.044$ (h) $P(z) = 0.025$

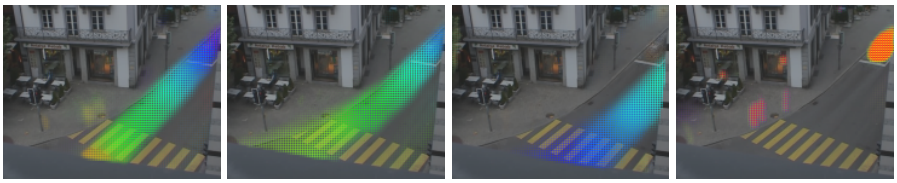
(i) $P(z) = 0.040$ (j) $P(z) = 0.039$ (k) $P(z) = 0.035$ (l) $P(z) = 0.040$

Fig. 16: MIT data. Representative motifs of 10s duration out of 26. (a–d) Activities due to waiting objects. (e–l) Activities due to motion. Best viewed in color. Please see the additional material [1] to view animated GIF versions of the motifs.

signal. Interestingly, another set of motifs capture pedestrian activities, despite the fact that they are less constrained and have more variability in localization, size, shape, timing and dynamics. This comprises people moving on the sidewalk (Fig. 17(e,f)), but also pedestrians crossing the road on the zebra crossing as in motifs from Fig. 17(g,h).

Metro Station data. Despite the disorderly movements that continuously occur in the scene, the method is able to capture typical structured motion patterns that exist. Typical activities consists of people moving towards the top of the scene from different origins due to their arrival from different places in the metro station. Fig. 18(a–c) show these movements from bottom right, bottom middle and bottom left respectively. Other typical activities correspond to people leaving the ticket machine towards the turnstiles Fig. 18(d) and crossing the turnstiles at different places Fig. 18(e,f), and mixed motion patterns, Fig. 18(g,h). This clearly demonstrates that our model can successfully extract patterns even from crowded scenes containing unstructured movements.

Motif duration. We also experimented with longer motif duration T_z. For instance Fig. 19 shows motifs of 20 second duration from all the three datasets. Since longer motifs capture more activities, the BIC measure selected only 16, 16, 14 and 12 motifs for the Far-field, MIT, Traffic junction and Metro station data respectively. Broadly speaking, when one extends the motif maximal length beyond the actual duration of a scene activity, the same motif is estimated, as already observed



(a) $P(z) = 0.090$ (b) $P(z) = 0.089$ (c) $P(z) = 0.085$ (d) $P(z) = 0.073$

(e) $P(z) = 0.058$ (f) $P(z) = 0.052$ (g) $P(z) = 0.052$ (h) $P(z) = 0.048$

Fig. 17: Traffic Junction data. Representative motifs of 10s duration. (a-d) vehicle activities. (e-h) pedestrian activities. Best viewed in color. Please see the additional material [1] to view animated GIF versions of the motifs.

with synthetic data[8]. This is typically the case with the short duration motifs due to vehicles in MIT (Fig. 16(f,l)) or Traffic junction (Fig. 17(a-c)) datasets. Still, as activities can often be described with different time granularities, variations or other motifs may appear. For instance, as the travel time of vehicles in the Far-field or MIT scenes usually lasts longer than 10 seconds, vehicle activities are now captured as a single motif as shown in Fig. 19 rather than as a sequence of shorter motifs of 5 to 10 seconds in length. As an example, the motif in Fig. 19(a) combines the activities of Fig. 15(e,d). The same applies with the pedestrian activities in the Traffic Junction case (cf Fig. 19(i,j)). In case of metro station data, the motif in Fig. 19(k) captures people moving towards the turnstile and crossing it, one after another. Fig. 19(l) shows movement from bottom of the scene.

Effect of MAP prior on motifs. In section 6.1, we proposed to use MAP to favor the occurrences of the motif words at the start of the motif. Since our motifs aim to capture real world activities, we prefer that the SLA activity starts exactly at the first time instant of the motif rather than after a few time steps. This is explained in Fig. 20.

In Fig. 20, the same activity of vehicle moving from bottom right and taking a right turn extracted from two runs of PLSM with $T_z = 10$ are shown. Fig. 20(a) is without a prior and Fig. 20(b) is with a MAP prior on t_r. The prior used is a decreasing ramp as a function of the time-step and the length of the motif, and given by $f(t_r) \propto (T_z - t_r) + c$. Fig. 20(c) and Fig. 20(d) show the $P(w, t_r|z)$ matrix of Fig. 20(a) and Fig. 20(b) respectively. The motifs are 10 time steps (10 seconds) long. Without prior in Fig. 20(a,c), we can observe that no words

[8] Note however that longer motifs increase the chance of observing some random co-occurrences, as the amount of overlap with other activities, potentially unexplained by current motifs, increases as well. This is particularly true when the amount of data is not very large like in the Traffic junction case.

(a) $P(z) = 0.038$ (b) $P(z) = 0.029$ (c) $P(z) = 0.013$ (d) $P(z) = 0.053$

(e) $P(z) = 0.072$ (f) $P(z) = 0.070$ (g) $P(z) = 0.048$ (h) $P(z) = 0.091$

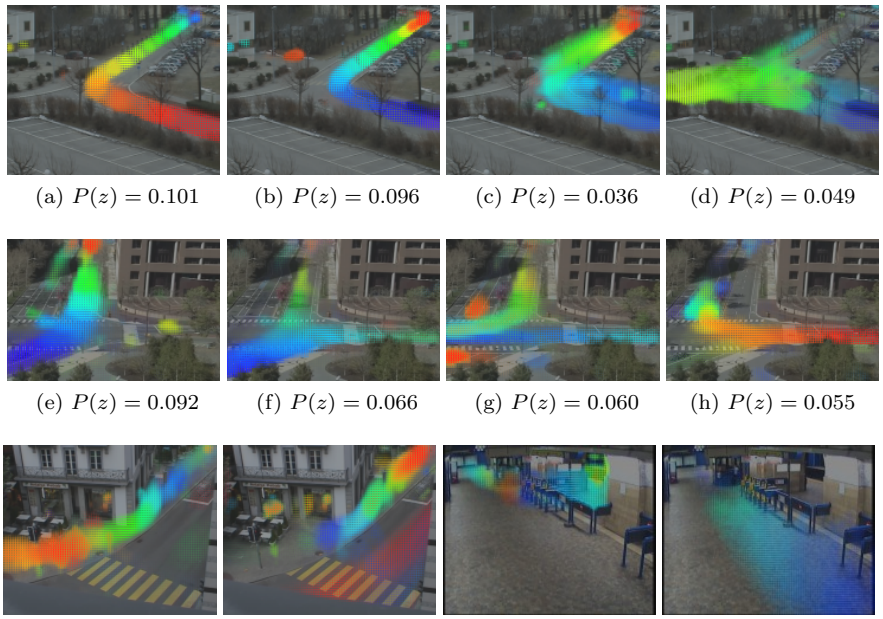
Fig. 18: Metro Station data. Eight representative motifs of 10s duration, out of 25 depicting (a–c) people moving towards top of the scene from bottom right, bottom-middle and bottom left respectively; (d) people leaving the ticket machine towards the turnstiles; (e–f) people around the turnstiles and crossing them at different places to reach the platform, and (g–h) mixed activities around the information booth and in the turnstiles area. Best viewed in color. Please see the additional material [1] to view animated GIF versions of the motifs.

occur in the first 3 time steps, which means that the activity captured by PLSM effectively starts after a relative time of 4 seconds from the beginning of the motif, as indicated by the rendered motif in Fig. 20(a) which starts with green color instead of blue. When using the ramp MAP prior on the temporal axis, the PLSM algorithm actually recovers motifs with words occurring in the first time steps of the motifs, as we can see in the matrix of Fig. 20(d) as well in Fig. 20(b) which starts with the Violet-Blue color. Notice that as a consequence of having words at the motif beginning, PLSM will better capture longer activities, as shown by the longer trailing parts in the matrix Fig. 20(d) which is reflected by a longer extension of the activity towards the top right of the scene in Fig. 20(b) image.

6.3 Event detection

To evaluate how well the recovered motifs match the real activities observed in the data, we performed a quantitative analysis by using the PLSM model to detect particular events. Indeed, as the model can estimate the most probable occurrences $P(t_s, z|d)$ of a topic z for a test document d, it is possible to create an event detector by considering all t_s for which $P(t_s, z|d)$ is above a threshold. By varying this threshold, we can control the trade-off between precision and completeness (*i.e.,* recall).

For this event detection task, we labeled a 12 minute video clip from the Far-field scene, distinct from the training set, and considered all the different car activities that pass through the three road junction. Activity categories that



(a) $P(z) = 0.101$ (b) $P(z) = 0.096$ (c) $P(z) = 0.036$ (d) $P(z) = 0.049$

(e) $P(z) = 0.092$ (f) $P(z) = 0.066$ (g) $P(z) = 0.060$ (h) $P(z) = 0.055$

(i) $P(z) = 0.102$ (j) $P(z) = 0.088$ (k) $P(z) = 0.161$ (l) $P(z) = 0.082$

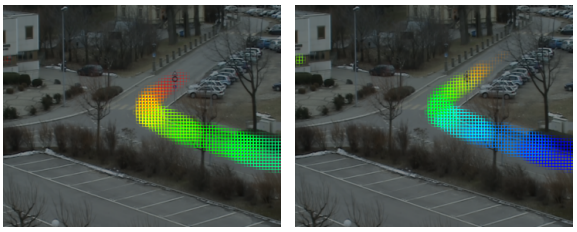
Fig. 19: Motifs with $T_z = 20$. Motifs of 20s duration that mainly differ from their 10s shorter counterparts. (a–d) Far-field, (e–h) MIT, and (i–j) Traffic junction data (k–l) Metro station data. All the above motifs capture the full extent of the activities within the scene. Best viewed in color. Please see the additional material [1] to view animated GIF versions of the motifs.

occurred fewer than 5 times in this test data were discarded, which left us with the 3 activity categories depicted in Fig. 21 with a total of 51 occurrences.

Given the fully unsupervised nature of our method, we manually associated each ground truth category to one of the discovered motifs (of maximum 10 second duration). This one to one manual association is a very minimal form of supervision and is somewhat suboptimal: the recovered PLSM motifs might not be matching the labelled events, and the one to one matching is somewhat limiting (see results below). The motifs considered for event detection are shown in 15(a,d,i). Using the occurrences $P(t_s, z|d)$ of these motifs[9], precision/recall curves were computed. They are shown in Fig. 21 both without and with sparsity.

From the curves, it is evident that for two out of the three events, we obtain a close to 100% result, especially with sparsity. The worst performance is for the activity "top right to bottom right" which gives a F-score of around 80%. This lower performance is due to the fact that two motifs can actually explain the same ground truth activity (they could be merged to improve results). Sparsity on the starting times exacerbates this effect by removing low probability detections.

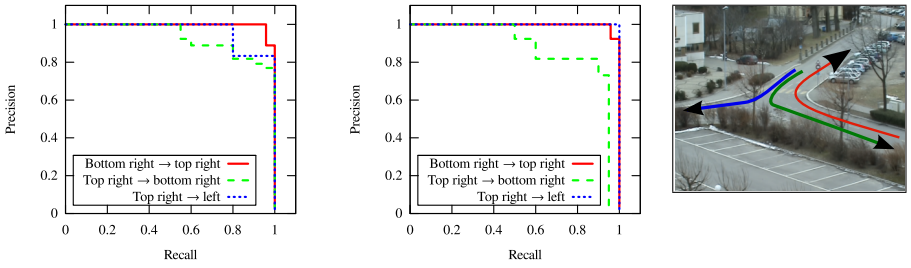
[9] To perform the temporal association we allowed a constant offset (either -1, 0 or 1) between the annotated instants of the event in the ground truth and the starting time of a learned motif.



(a) without MAP prior (b) with MAP prior

(c) $P(w, t_r|z)$ of (a) (d) $P(w, t_r|z)$ of (b)

Fig. 20: Illustration of the MAP prior effect on the recovered motifs. (a,c) motif obtained without applying a MAP prior; (b,d) corresponding motif to (a,c) obtained when applying the proposed temporal MAP prior on motifs. As we see in the later case, there is activity going on from the very beginning of the motif (*i.e.,* for low tr values), as compared to without a MAP prior.



(a) without sparsity (b) with sparsity (c) events

Fig. 21: Precision/recall curves for the detection of 3 types of events mapped onto 3 motifs, evaluated on a 12 minute test video. Results are provided both without and with sparsity. F-score at equal precision and recall (three curves): without sparsity (0.958, 0.818, 0.833), with sparsity (0.958, 0.818, 1). Area under the curves: without sparsity (0.995, 0.931, 0.967), with sparsity (0.997, 0.867, 1).

Overall, the results prove that the discovered motifs match the real activities well, and that motif starting times could be exploited for real event detection.

6.4 Activity prediction

The predictive model. The learned PLSM model can be used for predicting

the most probable future words. We have thus defined our task as estimating the probability $P_t^{pred}(w)$ that a word w appear at time t given all past information, that is, given the temporal document $n(w, t_a, d)$ up to time $t_a = t - 1$.

In our generative modeling approach, a word at time t can occur due to either a motif that has already started at a past time $t_s \in [t - T_z + 1, t - 1]$, or due to a motif that starts at the same time t. Hence, we define the prediction model as:

$$P_t^{pred}(w) \propto (1-\gamma) \sum_{t_s=t-T_z+1}^{t-1} \sum_z \hat{P}(t_s, z|d) P(w, t-t_s|z) + \gamma \sum_z P(z) P(w, 0|z), \tag{18}$$

where $\hat{P}(t_s, z|d)$ denotes our estimation that the motif z starts at time t_s given the observed data, γ represents the probability that a motif starts at the current instant, and $P(z)$ represents the motif prior probability estimated (along with the motifs) on training data[10]. To set γ, we have given equal priority to the starting time instants, and set $\gamma = \frac{1}{T_z}$, *i.e.,* a value of 0.1 in the current experiments. To obtain $\hat{P}(t_s, z|d)$ we simply apply our inference procedure to the temporal document $n(w, t_a, d)$ using only observations up to time $t - 1$.

Evaluation protocol and results. The prediction performance was evaluated using a standard 10 folds cross-validation approach. That is, each dataset (5500 and 6500 time steps in the MIT and Far-field cases, respectively) was split in 10 folds. Then, for each fold, the complementary 90% of the data was used to train a model that was tested and evaluated on this fold. The reported results are the average over the 10 folds. As performance measure, we used the average normalized prediction log-likelihood (ANL) defined as:

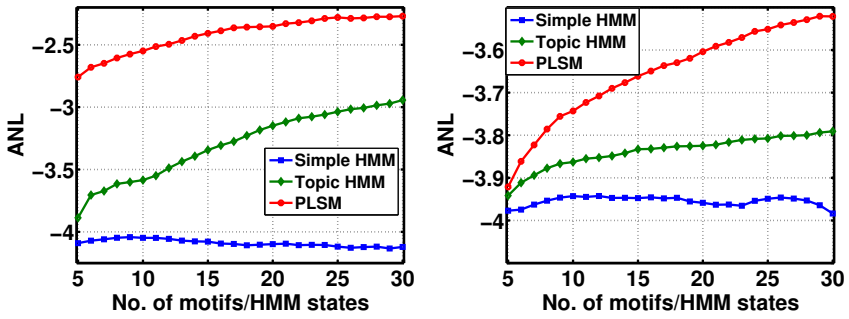
$$ANL = \frac{1}{N_{test}} \sum_t \frac{\sum_w n(w, t, d) \log(P_t^{pred}(w))}{\sum_w n(w, t, d)} \tag{19}$$

It is a standard measure for evaluating the modeling performance of topic models [46, 41], and is directly (inversely) related to the perplexity measure that is also commonly used to evaluate the generalization power of topic models [17, 5]. A higher ANL value indicates a better predictive capacity and vice versa. In order to compare the prediction accuracy of our model, we implemented two other temporal models.

Simple HMM. Here, the sequences of observation vectors $o_t(w) = n(w, t, d)$ from the training temporal documents were used to learn in an unsupervised fashion (*i.e.,* by maximizing the data-likelihood) a fully-connected HMM with n states. The emission probabilities were defined as Gaussians with a diagonal covariance matrix. At test time, the trained HMM was used to compute the expected state probability at time t given all observations up to time $t - 1$, from which the expected observation vector (and hence a predicted word probability $P_t^{pred}(w)$) was inferred.

Topic HMM. The second model is a more sophisticated approach in line with [18],

[10] Note that rather than simply using $P(z)$ as the prior for a topic to start at time t, we could have further exploited the past informations available in the past motif occurrences $\hat{P}(t_s, z|d)$ (*e.g.,* the motif of Fig. 15(e) is often followed by that of Fig. 15(d) several seconds later). However, as this is not part of our model, we preferred to go for the simpler case.



(a) Far-field data (b) MIT data

Fig. 22: Average Normalized Prediction log-likelihoods for (a) Far-field data, (b) MIT data. In both plots, the x-axis represents either the number of motifs (PLSM model), or the number HMM states in the two other cases.

wherein the Markov chain models the dynamics of a global behavior state. More precisely, we first apply PLSA (with n topics) to the set of training documents $\{o_t, t \in training\}$. This results in a set of topics $P(w|z)$ and topic distributions $o'_t(z) = P(z|o_t)$. We then learn an HMM with n states using the topic observation sequence o'_t. The HMM states learned with this method capture distinct scene level behaviors characterized by interacting topics and the Markov chain models the temporal dependencies among them. We thus refer to this method as *Topic HMM*. At test time, the expected state, topic and word probability distributions can be successively computed using the learned model.

Fig. 22 presents the results of PLSM and the two competitive methods. We observe that the simple HMM method gives the worst predictions on both datasets compared to the more sophisticated Topic-HMM, whose observations come from the PLSA topics. However, overall, the PLSM model gives a much better performance than the two HMM based methods, showing that the incorporation of temporal information at the motif level rather than at the global scene level is a better strategy. In the Far-field case, where the scene is not governed by any specific rules, PLSM performs consistently and significantly better with an average likelihood 200% greater than that of the Topic-HMM when $n = 5$, and 90% greater when $n = 30$. Note that both methods improve as more motifs or states are used, but saturates beyond a value of 30.

On the MIT data, the situation is somewhat different. PLSM and HMM based approaches perform similarly when the motifs/states is $n = 5$. The HMM approaches have an advantage as they are able to model the different phases of the regular cycle governed by the traffic lights that the scene goes through. These distinct global behavior states[11], and the transitions between them, are captured explicitly in the Topic-HMM and to a lesser extent in the HMM method whereas our method does not have any prior on the sequences of motif occurrences. Nevertheless, PLSM provides a finer and more detailed description of the activities and

[11] By global behavior state, we mean a state that control the set of activities that can occur in the whole scene *i.e.,* at a global scene level. For example, in the MIT video, the traffic lights turning red or green determines which activities can occur in the scene as a whole (vehicles will stop in one side of the road, while stopped vehicles start to move on the other side).

its prediction accuracy improves consistently beyond the performance of the other methods that have difficulties to take advantage of the modeling of questionable and unpredictable sub-phase global scene activity patterns. Note however that the difference with the other models is not as high in this case as on the Far-field data, but PLSM model still performs 35% better than the Topic-HMM. Finally, it is interesting to note that the prediction accuracy of the PLSM method tends to saturate for a number of motif N_z close to that selected using the BIC criterion (20 for the Far-field data, 26 for MIT data).

7 Discussion

The previous sections have shown that our model and approach was able to well recover dominant activities from long term observations that matched actual scene content and could be used for prediction. In the following, we discuss several elements that could be improved or would need modification to be applied to other datasets.

Vocabulary construction. Improving the accuracy or details of the captured activities for the videos could be achieved by incorporating features such as motion magnitude and foreground blob size. Such features are useful when there are a variety of objects appearing in the scene with different sizes and motion characteristics, as was shown for instance in [41]. However, scene-specific explicit calibration steps might be needed to correct the observed speed and size with respect to the object's distance from the camera. More importantly, they result in an increased vocabulary size that need to be dealt with: over-quantization of low-level features may result in instances of an activity being represented by quite different word distributions, making the separation between consistent repeated word co-occurrences and spurious ones more difficult.

Maximum motif duration. The maximum motif duration is currently set with some prior knowledge about the expected duration of the activities to be captured. In fact, from our synthetic data experiments we saw that setting this maximum duration to a larger value what is expected is a good strategy, as it limits the chances of breaking a recurring existing activity that would last longer, while causing no problem in identifying shorter activities. This was again observed in real life data, where we obtained consistent results when T_z was set to 10s and 20s. Nevertheless, finding a way to estimate this number automatically for each motif would be preferable and is a work in progress.

Comparison with [10]. We have used the Bayesian Information Criteria measure to determine the number of motifs. Alternatively, this could be dealt with by using other data driven approaches like [35,10]. For instance, our work [10] presents a method to extract and locate activity motifs that is qualitatively similar to this paper. There, motifs are represented in the same way, but Dirichlet processes at multiple levels are used to infer the appropriate number of motifs and their occurrences automatically. The method uses a more involved Gibbs sampling inference as compared to our more straightforward EM procedure. It is interesting to note that thanks to the Dirichlet process on the motif occurrences, the motif start time distributions are intrinsically sparse, precluding the necessity of an explicit sparsity constraint. Furthermore, due to this Dirichlet process, the posterior

probability for a motif to occur in a document is proportional to the number of times it has already occurred, which differs from our current model where it is proportional to the number of words attached to all occurrences of this motif (see Eq. 8). As a consequence, smaller motifs (that is, motifs generating less words) are treated more equivalently to larger ones with the Dirichlet approach. Although on our surveillance data we could not observe a significant difference in favor of one or the other method, it is a point that could affect the results on other datasets.

Activity timing variability. Our model handles local variations in local activity execution timing well but can cope only to a certain extent with differences in overall execution speed. If invariance to such speed is needed, there are several ways to handle it. For instance, we can conduct an a-posteriori analysis, by identifying motif replicas differing by speed execution variations. Or one can introduce an explicit latent variable to model the execution speed. Note that although this can be added in a straightforward manner in our model, this would result in increased computational complexity.

Higher-level model. Finally, our model identifies activities and their starting times, but has no statiscal model of the motif occurrence at the higher-level. The analysis and modeling of these occurrences in terms of dependencies or interactions could enhance the global understanding of the scene through, for instance, the identification of scene level rules (*e.g.,* right of way) or activity cycles due to the presence of a traffic light, as shown for instance in our recent results [39].

8 Conclusion

In this paper we proposed a novel unsupervised approach for discovering dominant activity motifs from multivariate temporal sequences. Our model infers sequential patterns of a maximum time duration by modeling the temporal co-occurrence of visual words, which significantly differs from previous topic model based approaches. This is made possible thanks to the introduction of latent variables representing the motif start times, bringing the following advantages: a) they help in implicitly aligning occurrences of the same motif while learning, and b) they allow us to infer when an activity starts. The model parameters can be inferred efficiently using an Expectation-maximization procedure that exploits a novel sparsity constraint. The effectiveness of our model was extensively validated using synthetic as well as real life data sets from both structured and unstructured scenes. Qualitative results and quantitative experiments on event detection and prediction tasks showed that the approach was discovering motifs consistent with the scene activities and was resulting in superior performance compared to other state of the art Dynamic Bayesian Network based alternatives.

A Appendix: detailed EM equation derivation

In this section we detail the derivation of equations involved in PLSM inference. Our data $\mathcal{D} = n(w, t_a, d)$ is a matrix of counts, where each triplet (w, t_a, d) denotes the number of times a word w, appears at time t_a in document d. The probability of observing this data is given by:

$$P(\mathcal{D}) = \prod_{d=1}^{D} \prod_{t_a=1}^{T_d} \prod_{w=1}^{N_w} P(w, t_a, d)^{n(w,t_a,d)} \tag{20}$$

Taking the log on both sides we obtain our objective log-likelihood function as

$$\mathcal{L}(\mathcal{D}) = \log P(\mathcal{D}) = \sum_{d=1}^{D}\sum_{t_a=1}^{T_d}\sum_{w=1}^{N_w} n(w,t_a,d)\log P(w,t_a,d) \tag{21}$$

Since $P(w,t_a,d) = \sum_{t_s=1}^{T_{ds}}\sum_{z=1}^{N_z} P(w,t_a,d,z,t_s)$, the log-likelihood equation conditioned on the model parameters Θ *i.e.,* the probability distributions $P(z|d)$, $P(t_s|z,d)$, and $P(w,t_r|z)$ is written as

$$\mathcal{L}(\mathcal{D}|\Theta) = \sum_{d=1}^{D}\sum_{w=1}^{N_w}\sum_{t_a=1}^{T_d} n(w,t_a,d)\log\sum_{z=1}^{N_z}\sum_{t_s=1}^{T_{ds}} P(w,t_a,d,z,t_s) \tag{22}$$

We want to infer the model parameters with a constraint that the distribution on motif starting times $P(t_s|z,d)$ is sparse and peaky. To achieve this goal as said in section 3.4, we want to maximize $D_{KL}(U||P(t_s|z,d))$, the KL divergence between Uniform distribution and $P(t_s|z,d)$. So the constrained log-likelihood equation is given by:

$$\mathcal{L}_c(\mathcal{D}|\Theta) = \mathcal{L}(\mathcal{D}|\Theta) + \sum_{z,d}\lambda_{z,d} D_{KL}(U||P(t_s|z,d)) \tag{23}$$

$$= \mathcal{L}(\mathcal{D}|\Theta) + \sum_{z,d}\lambda_{z,d}\sum_{t_s=1}^{T_{ds}}\frac{1}{T_{ds}}\log\frac{1/T_{ds}}{P(t_s|z,d)} \tag{24}$$

$$= \mathcal{L}(\mathcal{D}|\Theta) + \sum_{z,d}\lambda_{z,d}\left(\sum_{t_s}^{T_{ds}}\frac{1}{T_{ds}}\log\frac{1}{T_{ds}} - \sum_{t_s=1}^{T_{ds}}\frac{1}{T_{ds}}\log P(t_s|z,d)\right) \tag{25}$$

By removing the constant factor we obtain Eq. 5 given by

$$\mathcal{L}_c(\mathcal{D}|\Theta) = \mathcal{L}(\mathcal{D}|\Theta) - \sum_{t_s,z,d}\frac{\lambda_{z,d}}{T_{ds}}\log P(t_s|z,d) \tag{26}$$

But the above equation cannot be solved directly due to summation terms inside the log making it intractable. Instead, the EM approach works by optimizing the expected log-likelihood of the complete data with respect to the hidden variables keeping the constraint unchanged, which gives

$$E[\mathcal{L}_c] = \sum_{d=1}^{D}\sum_{w=1}^{N_w}\sum_{t_a=1}^{T_d}\sum_{z=1}^{N_z}\sum_{t_s=1}^{T_{ds}} n(w,t_a,d)P(z,t_s|w,t_a,d)\log P(w,t_a,d,z,t_s) - \sum_{t_s,z,d}\frac{\lambda_{z,d}}{T_{ds}}\cdot\log(P(t_s|z,d)) \tag{27}$$

Notice that now, the log operates over the joint probability over all the variables and not just the observed variables. An optimized way to compute this is by using Eq. 2 and indices t_r and t_s instead of t_a. The joint distribution can be split into its constituent distributions using Eq. 2 So, the expected log-likelihood equation is re-written as:

$$E[\mathcal{L}_c] = \sum_{d=1}^{D}\sum_{w=1}^{N_w}\sum_{t_r=1}^{T_z}\sum_{z=1}^{N_z}\sum_{t_s=1}^{T_{ds}} n(w,t_a,d)P(z,t_s|w,t_s+t_r,d)\log[P(z|d)P(t_s|z,d) P(t_r|z)P(w|t_r,z)] - \sum_{t_s,z,d}\frac{\lambda_{z,d}}{T_{ds}}\cdot\log P(t_s|z,d)$$

$$= \sum_{d=1}^{D}\sum_{w=1}^{N_w}\sum_{t_r=1}^{T_z}\sum_{z=1}^{N_z}\sum_{t_s=1}^{T_{ds}} n(w,t_a,d)P(z,t_s|w,t_s+t_r,d)[\log P(z|d) + \log P(t_s|z,d) + \log P(t_r|z) + \log P(w|t_r,z)] - \sum_{t_s,z,d}\frac{\lambda_{z,d}}{T_{ds}}\cdot\log P(t_s|z,d)$$

The goal is thus to optimize this expression with constraints so that the distributions sum to one. Therefore such constraints are enforced using lagrangian multipliers. Finally, the constrained objective function that is optimized is given by:

$$\mathcal{H}(\Theta) = E[\mathcal{L}_c] + \sum_z \gamma_z\left(1-\sum_{w,t_r}P(w,t_r|z)\right) + \sum_{z,d}\delta_{z,d}\left(1-\sum_{t_s}P(t_s|z,d)\right) + \sum_d \tau_d\left(1-\sum_z P(z|d)\right) \tag{28}$$

Where γ_z, $\delta_{z,d}$ and τ_d are the lagrangian multipliers. The EM algorithm works by iterating through the following E-step and M-step:

E-step: In the Expectation step, the posterior distribution of hidden variables (z, t_s) is computed where the parameters come from the previous iteration's M-step,

$$\begin{aligned} P(z,t_s|w,t_a,d) &= \frac{P(d,z,t_s,t_a,w)}{P(w,t_a,d)} \\ &= \frac{P(z|d)P(t_s|z,d)P(t_r|z)P(w|t_r,z)}{\sum_{z',t'_s}P(z'|d)P(t'_s|z',d)P(t_a-t'_s|z')P(w|t_a-t'_s,z')} \end{aligned} \tag{29}$$

M-step: In the Maximization step, maximizing $\mathcal{H}(\Theta)$ w.r.t to the parameters which are the probability mass functions results in the following set of equations.

$$\sum_{t_s=1}^{T_{ds}}\sum_{t_r=0}^{T_z-1}\sum_{w=1}^{N_w} n(w,t_s+t_r,d)P(z,t_s|w,t_s+t_r,d) - \tau_d P(z|d) = 0, \quad 1\le d\le D,$$

$$\sum_{t_a=1}^{T_d}\sum_{w=1}^{N_w} n(w,t_s+t_r,d)P(z,t_s|w,t_s+t_r,d) - \frac{\lambda_{z,d}}{T_{ds}} - \delta_{z,d}P(t_s|z,d) = 0, \quad 1\le d\le D, 1\le z\le N_z$$

$$\sum_{d=1}^{D}\sum_{t_s=1}^{T_{ds}} n(w,t_s+t_r,d)P(z,t_s|w,t_s+t_r,d) - \gamma_z P(w,t_r|z) = 0, \quad 1\le z\le N_z$$

by eliminating the lagrangian multipliers [12], we obtain the following expressions that were presented in (Eqs. 8 to 10)

$$P(z|d) \propto \sum_{t_s=1}^{T_{ds}}\sum_{t_r=0}^{T_z-1}\sum_{w=1}^{N_w} n(w,t_s+t_r,d)P(z,t_s|w,t_s+t_r,d) \tag{30}$$

[12] also taking into account that probabilities need to be positive. As a technical detail, a minimum value ε is required for $p(t_s|z,d)$ to avoid undefined log-likelihoods in the objective function.

$$P(t_s|z,d) \propto \max\left(\varepsilon, \left(\sum_{w=1}^{N_w}\sum_{t_r=0}^{T_z-1} n(w,t_s+t_r,d)P(z,t_s|w,t_s+t_r,d)\right) - \frac{\lambda_{z,d}}{T_{ds}}\right) \quad (31)$$

$$P(w,t_r|z) \propto \sum_{d=1}^{D}\sum_{t_s=1}^{T_{ds}} n(w,t_s+t_r,d)P(z,t_s|w,t_s+t_r,d) \quad (32)$$

References

1. http://www.idiap.ch/paper/plsm/plsm.html
2. Besnerais, G., Bercher, J., Demoment, G.: A new look at entropy for solving linear inverse problems. IEEE Trans. on Information Theory **45**(5), 1565–1578 (1999)
3. Blei, D., Lafferty, J.: A correlated topic model of science. Annals of Applied Statistics **1**(1), 17–35 (2006)
4. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: International Conference on Machine Learning, pp. 113–120 (2006)
5. Blei, D.M., Ng, A., Jordan, M.: Latent Dirichlet allocation. Journal of Machine Learning Research (3), 993–1022 (2003)
6. Boiman, O., Irani, M.: Detecting irregularities in images and in video. International Journal of Computer Vision **74**(1), 17–31 (2007)
7. Bradley, D., Bagnell, J.A.D.: Differentiable sparse coding. In: Proceedings of Neural Information Processing Systems 22 (2008)
8. Chen, S.S., Gopalakrishnan, P.S.: Speaker, environment and channel change detection and clustering via the bayesian information criterion. In: DARPA Broadcast News Transcription and Understanding Workshop, pp. 127–132 (1998)
9. Chien, J.T., Wu, M.S.: Adaptive bayesian latent semantic analysis. IEEE Transactions on Audio, Speech, and Language Processing **16**(1), 198–207 (2008)
10. Emonet, R., Varadarajan, J., Odobez, J.M.: Extracting and locating temporal motifs in video scenes using a hierarchical non parametric bayesian model. In: IEEE Conference on Computer Vision and Pattern Recognition (2011)
11. Fablet, R., Bouthemy, P.: Statistical motion-based object indexing using optic flow field. In: IEEE International Conference on Pattern Recognition, *ICPR*, vol. 4 (2000)
12. Faruquie, T.A., Kalra, P.K., Banerjee, S.: Time based activity inference using latent Dirichlet allocation. In: British Machine Vision Conference. London, UK (2009)
13. Girolami, M., Kabán, A.: On an equivalence between PLSI and LDA. In: ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 433–434 (2003)
14. Gohr, A., Hinneburg, A., Schult, R., Spiliopoulou, M.: Topic evolution in a stream of documents. In: SIAM International Conference on Data Mining, pp. 859–870 (2009)
15. Gruber, A., Rosen-Zvi, M., Weiss, Y.: Hidden topic Markov model. In: International Conference on Artificial Intelligence and Statistics. San Juan, Puerto Rico (2007)
16. Hervieu, A., Bouthemy, P., Cadre, J.P.L.: A statistical video content recognition method using invariant features on object trajectories. IEEE Transactions on Circuits and Systems for Video Technology **18**(11), 1533–1543 (2008)
17. Hofmann, T.: Unsupervised learning by probability latent semantic analysis. Machine Learning **42**, 177–196 (2001)
18. Hospedales, T., Gong, S., Xiang, T.: A Markov clustering topic model for mining behavior in video. In: IEEE International Conference on Computer Vision. Kyoto, Japan (2009)
19. Hospedales, T., Gong, S., Xiang, T.: Video behaviour mining using a dynamic topic model. International Journal of Computer Vision **98**(3), 303–323 (2012)
20. Hoyer, P.O.: Non-negative matrix factorization with sparseness constraints. Journal of Machine Learning Research **5**(2), 1457–1470 (2005)
21. Keogh, E., Chakrabarti, K., Pazzani, M., Mehrotra, S.: Dimensionality reduction for fast similarity search in large time series databases. Journal of Knowledge and Information Systems pp. 263–286 (2000)

22. Kuettel, D., Breitenstein, M.D., Gool, L.V., Ferrari, V.: What's going on? discovering spatio-temporal dependencies in dynamic scenes. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1951–1958 (2010)
23. Li, J., Gong, S., Xiang, T.: Global behaviour inference using probabilistic latent semantic analysis. In: British Machine Vision Conference (2008)
24. Li, J., Gong, S., Xiang, T.: Discovering multi-camera behaviour correlations for on-the-fly global activity prediction and anomaly detection. In: IEEE International Workshop on Visual Surveillance. Kyoto, Japan (2009)
25. Luvison, B., Chateau, T., Sayed, P., Pham, Q.C., Laprest, J.T.: An unsupervised learning based approach for unexpected event detection. In: International Conference on Computer Vision Theory and Applications (VISAPP), Lisboa, pp. 506–513 (2009)
26. Makris, D., Ellis, T.: Automatic learning of an activity-based semantic scene model. IEEE International Conference on Advanced Video and Signal Based Surveillance **2**(1), 183–188 (2003)
27. Mueen, A., Keogh, E., Zhu, Q., Cash, S., Westover, B.: Exact discovery of time series motifs. In: SIAM International Conference on Data Mining, pp. 473–484 (2009)
28. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. International Journal of Computer Vision **79**(3), 299–318 (2008)
29. Quelhas, P., Monay, F., Odobez, J.M., Gatica-perez, D., Tuytelaars, T.: A thousand words in a scene. IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(9), 1575–89 (2007)
30. Saleemi, I., Hartung, L., Shah, M.: Scene understanding by statistical modeling of motion patterns. In: IEEE Conference on Computer Vision and Pattern Recognition (2010)
31. Schwarz, G.: Estimating the dimension of a model. The Annals of Statistics **6**(2), 461–464 (1978)
32. Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A., Freeman, W.T.: Discovering object categories in image collections. In: IEEE International Conference on Computer Vision (2005)
33. Stauffer, C., L.Grimson, E.: Learning patterns of activity using real-time tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**, 747–757 (2000)
34. Tanaka, Y., Iwamoto, K., Uehara, K.: Discovery of time-series motif from multi-dimensional data based on MDL principle. Machine Learning **58**, 269–300 (2005)
35. Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Hierarchical Dirichlet processes. Journal of the American Statistical Association **101**(476), 1566–1581 (2006)
36. Tommasi, C., Kanade, T.: Detection and tracking of point features. Tech. rep., CMU-CS-91-132 (1991)
37. Tritschler, A., Gopinath, R.: Improved speaker segmentation and segments clustering using the bayesian information criterion. In: Sixth European Conference on Speech Communication and Technology (1999)
38. Varadarajan, J., Emonet, R., Odobez, J.: Probabilistic latent sequential motifs: Discovering temporal activity patterns in video scenes. In: British Machine Vision Conference, pp. 117.1–117.11. Aberystwyth (2010)
39. Varadarajan, J., Emonet, R., Odobez, J.: Bridging the Past, Present and Future; Modeling Scene Activities from Event Relationships and Global Rules. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
40. Varadarajan, J., Emonet, R., Odobez, J.M.: A sparsity constraint for topic models - application to temporal activity mining. In: NIPS Workshop on Practical Applications of Sparse Modeling: Open Issues and New Directions (2010)
41. Varadarajan, J., Odobez, J.: Topic models for scene analysis and abnormality detection. In: IEEE International Workshop on Visual Surveillance. Kyoto, Japan (2009)
42. Vasconcelos, N., Lippman, A.: Statistical models of video structure for content analysis and characterization. IEEE Transactions on Image Processing (1), 3–19 (2000)
43. Wallach, H.M.: Topic modeling: beyond bag-of-words. In: International Conference on Machine Learning, pp. 977–984. Pittsburgh, Pennsylvania (2006)
44. Wang, C., Blei, D.: Decoupling sparsity and smoothness in the discrete hierarchical Dirichlet process. In: Neural Information Processing Systems, pp. 1982–1989 (2009)
45. Wang, C., Blei, D.M., Heckerman, D.: Continuous time dynamic topic models. In: Conference on Uncertainty in Artificial Intelligence (2008)
46. Wang, X., Ma, X., Grimson, E.L.: Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. IEEE Transactions on Pattern Analysis and Machine Intelligence **31**(3), 539–555 (2009)

47. Wang, X., McCallum, A.: Topics over time: A non-Markov continuous-time model of topical trends. In: ACM Conference Knowledge Discovery and Data Mining. Philadelphia, USA (2006)
48. Wang, X., Tieu, K., Grimson, E.L.: Learning semantic scene models by trajectory analysis. In: European Conference on Computer Vision, vol. 14, pp. 234–778 (2004)
49. Wang, Y., Mori, G.: Human action recognition by semi-latent topic models. IEEE Transactions on Pattern Analysis and Machine Intelligence Special Issue on Probabilistic Graphical Models in Computer Vision **31**(10), 1762–1774 (2009)
50. Williamson, S., Wang, C., Heller, K., Blei, D.: Focused topic models. In: NIPS workshop on Applications for Topic Models: Text and Beyond. Whistler, Canada. (2009)
51. Xiang, T., Gong, S.: Video behavior profiling for anomaly detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **30**(5), 893–908 (2008)
52. Yang, Y., Liu, J., Shah, M.: Video scene understanding using multi-scale analysis. In: IEEE International Conference on Computer Vision. Kyoto, Japan (2009)
53. Yao, J., Odobez, J.M.: Multi-layer background subtraction based on color and texture. In: IEEE CVPR International Workshop on Visual Surveillance, pp. 1–8 (2007)
54. Yi Zhang Jeff Schneider, A.D.: Learning compressible models. In: Proceedings of SIAM Data Mining (SDM) Conference (2010)
55. Zhang, D., Gatica-Perez, D., Bengio, S., McCowan, I., G.Lathoud: Multimodal group action clustering in meetings. In: ACM International Conference on Multimedia, Workshop on Video Surveillance and Sensor Networks (2004)
56. Zhong, H., Jianbo, S., Mirko, V.: Detecting unusual activity in video. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 819–826. Washington, DC (2004)