Hierarchical Span-Based Conditional Random Fields for Labeling and Segmenting Events in Wearable Sensor Data Streams

Roy J. Adams RJADAMS@CS.UMASS.EDU
University of Massachusetts Amherst, 140 Governors Drv, Amherst, MA 01003

Nazir Saleheen NSLEHEEN@MEMPHIS.EDU
University of Memphis, 375 Dunn Hall, Memphis, TN 38152

Edison Thomaz ETHOMAZ@UTEXAS.EDU
The University of Texas at Austin, 110 Inner Campus Drive, Austin, TX 78705

Abhinav Parate APARATE@CS.UMASS.EDU
Lumme Inc, 34 Main St, Amherst, MA 01002

Santosh Kumar SKUMAR4@MEMPHIS.EDU
University of Memphis, 375 Dunn Hall, Memphis, TN 38152

Benjamin M. Marlin MARLIN@CS.UMASS.EDU
University of Massachusetts Amherst, 140 Governors Drv, Amherst, MA 01003

Abstract

The field of mobile health (mHealth) has the potential to yield new insights into health and behavior through the analysis of continuously recorded data from wearable health and activity sensors. In this paper, we present a hierarchical span-based conditional random field model for the key problem of jointly detecting discrete events in such sensor data streams and segmenting these events into high-level activity sessions. Our model includes higher-order cardinality factors and inter-event duration factors to capture domain-specific structure in the label space. We show that our model supports exact MAP inference in quadratic time via dynamic programming, which we leverage to perform learning in the structured support vector machine framework. We apply the model to the problems of smoking and eating detection using four real data sets. Our results show statistically significant improvements in segmentation performance relative to a hierarchical pairwise CRF.

Proceedings of the 33^{rd} International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48.

1. Introduction

A small number of behaviors including physical inactivity, poor diet, tobacco use, and alcohol consumption are key risk factors in a wide array of chronic conditions including obesity, cancer, diabetes and cardiovascular disease (McGinnis et al., 2002; Mokdad et al., 2004; DeVol et al., 2007). These behaviors have traditionally been studied using self-report data; however, self-report has well-known limitations including data sparsity, recall bias, and high burden on study subjects (Shiffman et al., 2008). The emerging field of mobile health (mHealth) seeks to replace the use of self report data with continuously recorded physiological and activity-related data streams collected using wearable sensors. While mHealth technologies have the potential to yield novel insights into health and behavior, significant data analysis challenges must first be overcome (Kumar et al., 2013).

In this paper, we address the key problem of detecting discrete events in wearable sensor data streams and segmenting these events into high-level activity sessions. This problem is central to several important mHealth tasks including smoking detection (Ali et al., 2012; Saleheen et al., 2015) and eating detection (Parate et al., 2014; Thomaz et al., 2015). In these domains, the events correspond to individual smoking puffs or eating gestures, the high-level activity sessions correspond to smoking a complete cigarette or eating a meal, and data streams may be available from a vari-

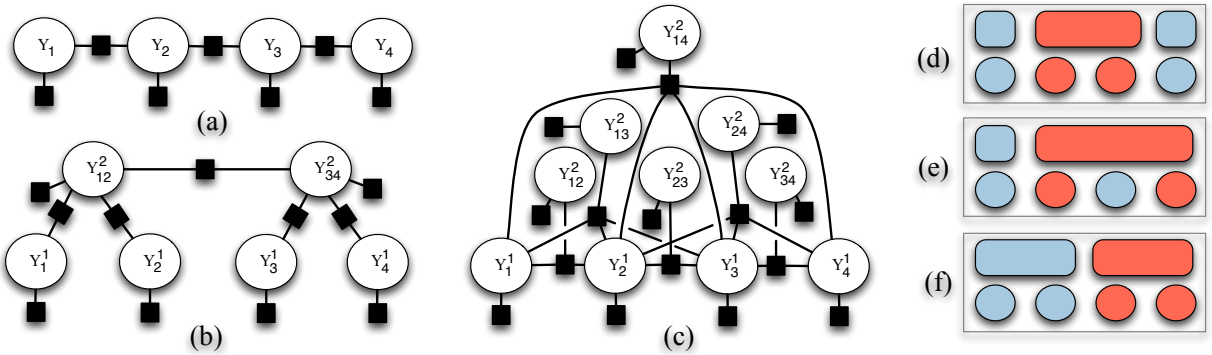


Figure 1. Figure (a) shows a factor graph model for a standard linear chain CRF over a length-four sequence. (b) shows a two-level hierarchical CRF where the first level labels are grouped into fixed size blocks at the second level, which has linear chain structure. Figure (c) shows a two-level version of the proposed model, which includes a quadratic number of second level span variables, one for each possible span. The global coordinating factor that ensures a valid segmentation connects to all second-level span variables and is not pictured. Figures (d)-(f) show example segmentations and labelings for a length four sequence. Our model (c) represents a distribution over both active spans and labels conditioned on the input features.

ety of sensors including respiration chest bands, wrist-worn actigraphy devices (smart watches), and other devices.

Since the underlying sensor waveforms in these domains are typically quasi-periodic, existing approaches to the event detection problem are based on performing an unsupervised segmentation of the raw sensor data stream into periods (respiration cycles, hand gestures, etc.), followed by the application of standard machine learning methods to independently classify each period as corresponding to an event of interest (smoking puff, eating gesture, etc.) or not (Ali et al., 2012; Saleheen et al., 2015; Parate et al., 2014; Thomaz et al., 2015).

By contrast, the segmentation problem is significantly more complex due to the fact that the segments can have arbitrary lengths, and the events that comprise a segment can have heterogeneous labels (e.g.: the respiration cycles that occur during a smoking segment are a mixture of both smoking puffs and non-puffs) (Ali et al., 2012; Saleheen et al., 2015). As a result, prior work within the mHealth research community has either ignored the session delineation problem completely, used methods based on ad-hoc post-processing of detections, or stacked simple segmentation models like linear chain conditional random fields (CRFs) on top of the detected events (Ali et al., 2012; Saleheen et al., 2015; Parate et al., 2014; Thomaz et al., 2015).

The primary contributions of this paper are the development and evaluation of a novel hierarchical span-based CRF model and a quadratic-time exact *maximum a posteriori* (MAP) inference algorithm that can solve the event detection and segmentation problems jointly. The proposed model incorporates higher-order factors to enforce a valid nested segmentation, and includes a variety of additional higher-order factors to enable the expression of key domain structure including the distribution of inter-event durations, and the number of events per segment. These factors make the proposed model significantly more expressive than standard pairwise CRF models used for segmentation in computer vision and other areas (Shotton et al., 2006). Additionally, this domain structure generalizes to multiple mHealth detection problems, obviating the need for custom or ad-hoc solutions.

The proposed inference algorithm is based on dynamic programming and is closely related to inference in semi-Markov CRFs (Sarawagi & Cohen, 2004). We leverage this MAP inference algorithm to learn the model parameters within the structured support vector machine (SSVM) framework (Tsochantaridis et al., 2005). We note that the model we propose may thus be equivalently viewed as a higher-order CRF or an SSVM. We choose to describe the model as a CRF and represent it graphically using a factor graph (Kschischang et al., 2001).

The remainder of the paper is organized as follows. In Section 2, we describe related work. In Section 3 we describe the proposed model and inference method. In Section 4, we present experiments on synthetic data and four real mHealth data sets covering two application domains: eating detection and smoking detection.

2. Related Work

In this section, we describe related discriminative structured prediction models. Linear-chain CRFs (see Figure 1(a)) were introduced by Lafferty et al. (2001) for structured prediction problems in natural language processing. They capture first-order dependencies between the labels in the sequence. CRF models with local pairwise dependencies have also been used in the computer vision literature to segment images (Shotton et al., 2006; Verbeek & Triggs, 2008).

The limitation to first-order local dependencies is clearly restrictive and more general models have subsequently been proposed. Skip-chain CRFs generalize linear chain models by allowing for longer-range pairwise dependencies between labels in the sequence (Sutton & McCallum, 2006, p.117). An alternative approach to inducing long range dependencies is to use a hierarchical CRF. A two-level example is shown in Figure 1(b). Multi-level versions of this type of model have been applied in the computer vision literature using a two-step procedure that first constructs a nested segmentation of an image using unsupervised methods, and then labels the fixed segmentation hierarchy (Reynolds & Murphy, 2007; Plath et al., 2009).

Our proposed modeling framework (see Figure 1(c)) is instead closer to the semi-Markov CRF model introduced by Sarawagi & Cohen (2004). Their approach includes a global factor for ensuring that only valid segmentations are considered by the model. Our model uses related global coordination factors to define a probability distribution over a three-level label hierarchy. Both our model and the semi-CRF model are able to define features on segments. However, the semi-CRF assumes that the labels within a segment are homogeneous, while our model allows restricted sequences of heterogeneous labels as described in the next section.

Our model is also related to the CRF context free grammar (CRF-CFG) model introduced by Finkel et al. (2008). Given a context free grammar specified by a collection of productions, the CRF-CFG framework associates weighted feature functions with each production. The productions, weights, and feature functions together induce a distribution over parse trees of an input sequence. The feature functions can depend on features of the input sequence spanned by the sub-tree rooted at the node where the production is applied, as well as the left and right arguments of a production.

While the CRF-CFG model can represent nested structures of the type we consider in this work using an appropriate grammar, the restricted local dependence of the CRF-CFG feature functions on the productions (labels) makes it cumbersome to incorporate higher-order factors. Of particular interest in our applications is the use of cardinality factors that count the number of labels of a given type that sit below a given segment. Our approach to incorporating these higher-order factors is closely related to the approach to handling such factors introduced by Smith & Eisner (2008).

In terms of inference, the semi-Markov CRF model supports quadratic-time exact MAP inference based on dynamic programming while producing a flat segmentation with homogeneous labels within segments (Sarawagi & Cohen, 2004). The CRF-CFG model supports exact MAP inference for the most likely parse tree via the inside-outside dynamic programming, but requires cubic time because it allows for more general structures (Finkel et al., 2008). The inference algorithm we propose is also based on dynamic programming and like the semi-Markov CRF it has quadratic complexity. However, the model we propose produces a multi-level segmentation with a restricted heterogeneous labeling within segments while incorporating other higher-order factors on segments, including cardinality factors.

Finally, related models have been proposed for activity recognition from video streams. Tang et al. (2012) and Sung et al. (2012) both propose models with a single layer of segmentation that allow for heterogeneous sequence labels beneath a segment; however, both models assume Markov structure within and between segments and do not model any higher-order structure.

3. Model, Inference and Learning

In this section we introduce our conditional random field model for hierarchical nested segmentation (HNS) of event sequences. As mentioned in the introduction, the sensor data streams of interest in this work are quasi-periodic and are pre-segmented into periods in an unsupervised manner as part of pre-processing the raw data. The model we introduce in this section assumes that each input stream is discretized into a sequence of n individual periods. In this work we focus on offline analysis of pre-recorded data streams. We refer to each point in the resulting discrete sequence as an **event**. For our application to mHealth, we define an HNS model with two segmentation layers: the first layer represents **inter-event** intervals and the second layer represents complete activity **sessions**.

In the case of smoking, the events are respiration cycles and the labels correspond to puffs and non-puffs, a positively labeled inter-event interval corresponds to an inter-puff interval, and a positively labeled session corresponds to the time span in which a single cigarette is smoked. In the case of eating, the events correspond to individual gestures and the labels are eating gestures and non-eating gestures. A positively labeled inter-event interval corresponds to the interval between eating gestures. A positively labeled session corresponds to the time span in which a single meal or snack is eaten. Importantly, smoking and eating sessions can contain both positive and negative labels below them, while non-smoking and non-eating session only contain negative labels.

3.1. Notation and Model Definition

The event layer of our proposed model consists of one label variable $Y_i^{(1)} \in \mathcal{Y}^{(1)} = \{0, 1\}$ for each position i in the length n input sequence. Each additional layer consists

of $\mathcal{O}(n^2)$ variables $Y_{jk}^{(l)} \in \mathcal{Y}^{(l)} = \{\emptyset, 0, 1\}$ that provide a label for a span of the input sequence starting at position j and ending at position k. $\mathcal{Y}^{(l)}$ is the set of possible labels at level l. Importantly, for $l > 1$, $\mathcal{Y}^{(l)}$ contains a special null label, $\emptyset$, denoting that a span is not used in the segmentation. This allows the model to define a joint probability distribution over active segments and labels. Additionally, we may have features available at any layer of the model. We denote the features associated with the base layer label variable $Y_i^{(1)}$ by $\mathbf{X}_i^{(1)}$ and the features associated with $Y_{jk}^{(l)}$ by $\mathbf{X}_{jk}^{(l)}$. As stated above, for activity segmentation we use the event level and two segmentation levels.

The joint probability of the label variables given the feature variables in a CRF is defined by a collection $\mathcal{F}$ of non-negative factor functions. Let $\mathbf{Y}$ represent the collection of all label variables in the model, $\mathcal{Y}$ represent the set of all possible joint labelings, $\mathbf{X}$ represent the collection of all feature variables, and θ represent the model parameters. The joint probability of an assignment to the label variables given the feature values is defined below where the partition function $Z_\theta(\mathbf{x})$ sums over all possible joint segmentations and labelings. The following sections describe the factors used in the proposed model.

$$P_\theta(\mathbf{Y} = \mathbf{y}|\mathbf{X} = \mathbf{x}) = \frac{1}{Z_\theta(\mathbf{x})} \prod_{\phi \in \mathcal{F}} \phi_\theta(\mathbf{y}, \mathbf{x}) \qquad (1)$$

3.2. Local Factors

We use local factors in each of the three levels of our model to incorporate features, enforce structural constraints, and model structural regularities.

Event Level: The first level labels in the model $Y_i^{(1)} \in \{0, 1\}$ indicate the event type for individual events. The first level features $X_i^{(1)}$ are generally extracted from the input stream spanned by event i. The model includes standard log-linear feature factors $\psi_i^{(1)}(y) = \exp(\mathbf{w}_y^{(1)} \mathbf{x}_i^{(1)})$ between the features and the labels. These features are task specific and depend on the sensing modalities used. We also constrain the labels on the bottom level to be negative if the labels in the second level are negative using a hard factor $\pi_i^{(1)}$.

$$\pi_i^{(1)} = \begin{cases} 0 & \text{if } Y_i^{(1)} = 1 \text{ and } \exists_{j \le i \le k} \text{ s.t. } Y_{jk}^{(2)} = 0 \\ 1 & \text{otherwise} \end{cases}$$

Inter-Event Level: The positive mid-level inter-event span variables $Y_{jk}^{(2)} \in \{0, 1, \emptyset\}$ are defined to start on a positive event label, and end before the next positive event label. In our applications, the feature vector $X_{jk}^{(2)}$ consists of a one-hot encoding of the duration of the binned inter-event intervals. These spans have standard log-linear feature factors $\psi_{jk}^{(2)}(y) = \exp(\mathbf{w}_y^{(2)} \mathbf{x}_{jk}^{(2)})$ between the features and the span labels.

The constraint that the label of the first event j beneath a positive inter-event interval $Y_{jk}^{(2)}$ must be positive is encoded via the hard label position factor $\pi_{jk}^{(2)}$. We also require that the labels on the inter-event intervals match those of the session spans they nest under as encoded by the factors $\Phi_{jk}^{(2)}$.

$$\pi_{jk}^{(2)} = \begin{cases} 1 & \text{if } Y_{ij}^{(2)} = 1, Y_i^{(1)} = 1, \text{and } \forall_{i<k\le j} Y_k^{(1)} = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\Phi_{jk}^{(2)} = \begin{cases} 1 & \text{if } Y_{jk}^{(2)} \neq \emptyset \text{ and } \exists_{i \le j, k \le l} \text{ s.t. } Y_{il}^{(3)} \neq \emptyset \\ & \quad \text{and } Y_{jk}^{(2)} = Y_{il}^{(3)} \\ 0 & \text{otherwise} \end{cases}$$

Session Level: At the top level of the model, $Y_{jk}^{(3)} \in \{0, 1, \emptyset\}$ labels the span starting at event j and ending at event k. These spans represent complete activity sessions. The valid labels are positive, negative, or null. In the case of smoking, if $Y_{jk}^{(3)} = 1$, then the model predicts a smoking session starting at event j and ended at event k. If $Y_{jk}^{(3)} = 0$, the model predicts a non-smoking session between events j and k. If $Y_{jk}^{(3)} = \emptyset$, the span from j to k is not used in the segmentation.

At the top level of the model, we include a cardinality factor $\kappa_{jk}^{(3)}$ on the count c of the number of positive event labels contained within the session (equivalent to the number of positive inter-event intervals). This factor is defined via a function $h(c; \omega) : \mathbb{N} \to \mathbb{R}$. We implement this function as a table look-up $h(c; \omega) = \omega_c$. The factor function is shown in detail below.

$$\kappa_{jk}^{(3)} = \exp\left[[Y_{jk}^{(3)} = 1] h \left(\sum_{j'=j}^{k-1} \sum_{k'=j'+1}^{k} [Y_{j'k'}^{(2)} = 1]; \omega \right) \right]$$

We also require that adjacent active session spans have opposite labels so that top level sessions are not fragmented into multiple spans. This is implemented via the within-level factor $\Omega_{ij}^{(3)}$ as shown below.

$$\Omega_{ij}^{(3)} = \begin{cases} 0 & \text{if } Y_{ij}^{(3)} \neq \emptyset \text{ and } \exists k \text{ st } Y_{ij}^{(3)} = Y_{j+1k}^{(3)} \\ 1 & \text{otherwise} \end{cases}$$

3.3. Global Factors

Enforcing the nested segmentation property requires two sets of high-order factors. First, every label variable $Y_i^{(1)}$ must be covered by exactly one non-null span variable $Y_{jk}^{(l)}$ at each level $l > 1$. This is enforced using a binary-valued factor $S^{(l)}$ as shown below where $\exists_{=1}$ means "there exists

$$\alpha^{(1)}(j,k) = \begin{cases} \psi_j^{(1)}(1) & \text{if } j = k \\ \alpha^{(1)}(j,k-1)\psi_k^{(1)}(0) & k > j \end{cases}$$

$$\alpha^{(2)}(j,c) = \begin{cases} 1 & \text{if } j = n+1 \\ \max\limits_{j \le k \le n} \max\limits_{c \in \{1,..,C\}} \alpha^{(1)}(j,k)\, \psi_{jk}^{(2)}(0) e^{\omega_c} \alpha^{(2)}(k+1,c) & \text{if } c = 0 \\ \max\limits_{j \le k \le n} \alpha^{(1)}(j,k)\, \psi_{jk}^{(2)}(1)\, \alpha^{(2)}(k+1,c-1) & \text{if } c > 0 \end{cases}$$

Algorithm 1. Dynamic program for inference in the HNS model.

exactly one". This ensures that the labeling at level l forms a valid segmentation with no overlapping segments and no gaps in the segmentation. There exists one such factor for each segmentation level in our model.

$$S^{(l)} = \begin{cases} 1 & \text{if } \forall\, i\ \exists_{=1}\, (j,k) \text{ s.t. } j \le i \le k \text{ and } Y_{jk}^{(l)} \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

$$N^{(l)} = \begin{cases} 1 & \text{if } \forall\, (j,k) \text{ s.t. } Y_{jk}^{(l)} \neq \emptyset,\ \exists p > j, q < k \\ & \quad \text{s.t. } Y_{jp}^{(l-1)} \neq \emptyset \text{ and } Y_{qk}^{(l-1)} \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

Second, to enforce the nesting property, we require that the span boundaries at level l align with the span boundaries at level $l-1$ for $l = 2, 3$. The factor $N^{(l)}$ shown above ensures that each endpoint of a non-null span variable will align with an endpoint of a non-null span variable in the layer below it. The complete set of factors defining the model is thus $\mathcal{F} = \{S^{(2)}, S^{(3)}, N^{(2)}, N^{(3)}, \psi_i^{(1)}, \psi_{jk}^{(2)}, \psi_{jk}^{(3)}, \pi_{jk}^{(1)}, \pi_{jk}^{(2)}, \Phi_{jk}^{(2)}, \kappa_{jk}^{(3)}, \Omega_{ij}^{(3)}\}$ for all $1 \le i \le n$, all $1 \le j \le k \le n$.

3.4. Inference

Due to the special nested structure of the model and the hard constraints imposed, it is possible to run exact MAP inference in the three layer HNS model in time quadratic in the length n of the input sequence. In Algorithm 1, we give the dynamic program recursions for computing the unnormalized joint probability of the MAP solution. The algorithm consists of two related recursions. The first recursion $\alpha^{(1)}(j,k)$ computes the energy contribution of all valid joint assignments to the bottom level labels $Y_j^{(1)}$ through $Y_k^{(1)}$, taking into account the features and inter-event span constraints.

The second recursion $\alpha^{(2)}(j,c)$ computes values for the inter-event intervals and the sessions simultaneously, taking into account the hard constraints and cardinality factor. The two indices are the position in the sequence j and the value for the cardinality factor c. The value $\alpha^{(2)}(j,c)$ corresponds to the unnormalized probability of the MAP segmentation of positions j through n starting with a segment of cardinality c ($c = 0$ implies a negative segment).

From top to bottom, the three cases in $\alpha^{(2)}(j,c)$ are the recursion base case, which holds if $j = n+1$, the case where we are in a top-level span with a negative label ($c = 0$), and the case where we are in a top-level span with a positive label and cardinality $c \ge 1$. When $c = 0$, the next session must be positive due to the $\Omega^{(3)}$ factor, so we must maximize over both the length of the current segment and the cardinality of the positive session that follows. We allow at most C positive events beneath a positive segment. When $c > 0$, the following inter-event segment must be the first of $c-1$ subsequent positive segments, so we need only maximize over the length of the current inter-event span.

Finally, we calculate $\alpha^{(3)} = \max\{\alpha^{(2)}(0,0), \max_{c \in \{1,..,C\}} e^{\omega_c} \alpha^{(2)}(0,c)\}$ and retrieve the MAP assignment as the path used to calculate $\alpha^{(3)}$. This final computation takes into account the fact that a sequence can start with either a positive or negative session. Running these dynamic programs has complexity $\mathcal{O}(n^2 C)$ where $C << n$. The computation of the feature function values $\psi_{jk}^{(l)}(y)$ for all spans jk can also be computed in quadratic time and cached, so the overall MAP inference procedure is quadratic in the length of the input sequence. This algorithm is similar in structure and complexity to the inference method for the semi-markov CRF (Sarawagi & Cohen, 2004), but allows for heterogeneous labels at the base level, incorporates the cardinality factor, and adds an additional layer to capture the distribution of inter-event durations.

3.5. Learning

We learn model parameters using large margin learning methods (Tsochantaridis et al., 2005), making our model equivalent to a structured support vector machine (SSVM). In the case of CRFs, the learning algorithm for SSVMs expands a working set of constraints on each iteration to include constraints derived from the MAP labeling of each data case based on the current parameters. However, as observed by Tsochantaridis et al. (2005), this method treats all margin violations equally. Instead, we would like to penalize poor segmentations more heavily. This can be done by scaling the margin constraints with a user defined loss function that corresponds to the sum of the hamming loss

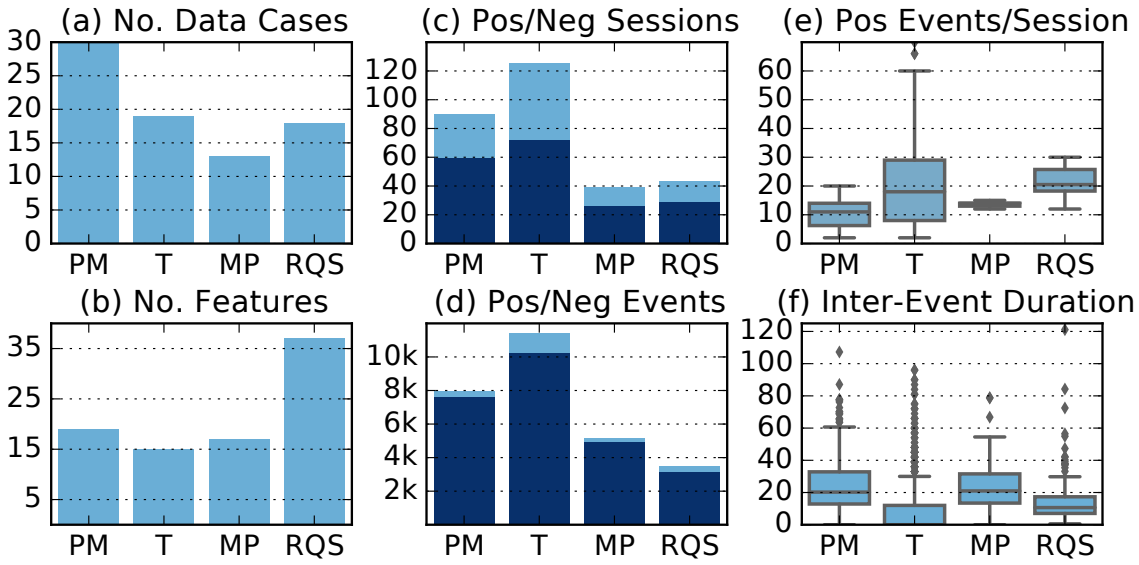


Figure 2. Summary statistics of the four datasets used. Events correspond to simple actions such as hand gestures and sessions correspond high level activities such as eating a meal. The subplots are: (a) the number of data cases, (b) the number of features, (c) the number of negative sessions (dark) and number of positive sessions (light), (d) the number of negative events (dark) and number of positive events (light), (e) the distribution of the number of positive events in a positive session, and (f) the distribution of the time in seconds between positive events. For display purposes, three outliers were omitted from the T box-plot in subplot (f).

between the true and predicted segmentation at each level when projected onto the bottom level of the model (Taskar et al., 2004a). Using this approach requires solving a loss-augmented MAP inference problem, which can be accomplished by adding two extra features to each feature vector as shown by Taskar et al. (2004b). The time complexity of loss augmented MAP inference thus remains quadratic.

Finally, we note that the dynamic programs presented can be extended to perform marginal inference. Consequently, other learning algorithms are possible including maximum likelihood learning. We focus on maximum margin training here as we have found that this approach results in better performance on both smoking and eating detection.

4. Experiments and Results

In this section, we describe the details of our data, tasks, experimental procedures, and results. All of the models described were implemented in Python and PyStruct was used for SSVM learning (Müller & Behnke, 2014).

Data and Tasks: We evaluate our model using synthetic data (described later), and four real mHealth datasets. The mPuff dataset (MP) was collected by Ali et al. (2012) and contains respiration data from smokers recorded using a chest band sensor. The puffMarker dataset (PM) was collected by Saleheen et al. (2015) and contains contains data from smokers recorded using a chest band sensor and a wrist-worn actigraphy device (accelerometer and gyroscope). The RisQ (RQS) dataset was collected by Parate et al. (2014) and contains wrist-worn actigraphy data. Finally, we use the data originally published by Thomaz et al. (2015) (T), which also contains wrist-worn actigraphy data for eating.[1] In all cases, we use the base-level discretization presented in the original papers. For each dataset, this results in a collection of discrete sequences which we call data cases. The PM dataset contained data cases that were to long to consider in a single model, so these were split into random sized pieces with each piece containing a single positive session. Summary statistics for each of the real datasets are shown in Figure 2. We can see that the data sets vary significantly in terms of their properties, including the degree of conservation of structure in the label space (Figure 2(e-f)).

There are two tasks of interest in these datasets. **Task 1** is to predict whether each event (respiration cycle or hand gesture) corresponds to a positive event (smoking puff or bite of food). **Task 2** is to segment the events into contiguous, non-overlapping sessions and to label each session as positive (a smoking or eating session) or negative.

Features: For each of the datasets described above, we use the event-level features originally published with the data with the exception of PM, where we omit actigraphy

[1]The Thomaz et al. (2015) dataset is available at `http://www.ethomaz.com/publications.html` and all other datasets are available from the original authors under appropriate data use agreements.

features used in the original paper which were not available for all events. Additionally, we apply a simple non-linear transformation to these features by finding five equal sized percentile bins for each feature and calculating the distance from the center of each percentile bin to the input feature value.

In addition to features at the event level, our model allows for the definition of features for spans. We include features intrinsically defined at the segment level such as the duration of the segment. In our model, the duration features for positive inter-event intervals correspond to the time between consecutive positive events within a positive session, a quantity that tends to be conserved in these applications.

Baselines: We compare the HNS model against two baselines: a Logistic Regression (LR) model, and the tree structured pairwise CRF (T-CRF) shown in Figure 1(b). The T-CRF model includes two levels of labels: session-level labels and event-level labels. Each session-level label corresponds to a window of events in the base sequence (Figure 1(b) shows a model with a window size of two). The window size is tuned as a hyper-parameter. The T-CRF model thus provides a strong segmentation baseline which allows for heterogeneous event labels beneath homogenous session labels, but is restricted to pairwise factors. We generate session-level features by averaging the event-level features sitting beneath each window. The LR model was trained using ℓ_2 regularized maximum likelihood and the T-CRF model was trained using ℓ_2 regularized maximum-margin methods. On Task 1 we compare against both the LR and T-CRF models and on Task 2 we compare only against the T-CRF model since LR does not produce an explicit segmentation.

Testing and Hyper-Parameter Selection: We conduct experiments using a stratified 10-fold cross-validation protocol. Specifically, we split the data cases into two groups, one for all data cases containing positive sessions and one for the rest, and randomize within groups. Next, we create 10 test folds so that each test fold contains approximately the same number of data cases from each group. To select hyper-parameter values, we perform a further stratified 10-fold cross-validation on the training samples. We use this cross-validation procedure for all methods. The LR hyper-parameters were tuned to maximize event-level F_1 score, while the T-CRF and HNS hyper-parameters were tuned to maximize segmentation accuracy. ℓ_2 regularization hyper-parameters were tuned on a logarithmic grid and all other hyper-parameters were tuned on a linear grid.

Performance Metrics: We assess performance for Task 1 (Labeling) using precision, recall, and F_1 score, which adjusts for the major class imbalance we face in this problem. We do not report accuracy due to strong class imbalance. We compute these metrics on each of the 10 test folds and report the mean scores as well as the standard error of the mean. For Task 2 (Segmentation), we compare the predicted sessions to the true sessions by projecting each session labeling onto the input sequence, and calculating the the precision, recall, and F_1 score of the projected labels. We report the mean of each segmentation metric over the 10 test folds as well as the standard error of the mean.

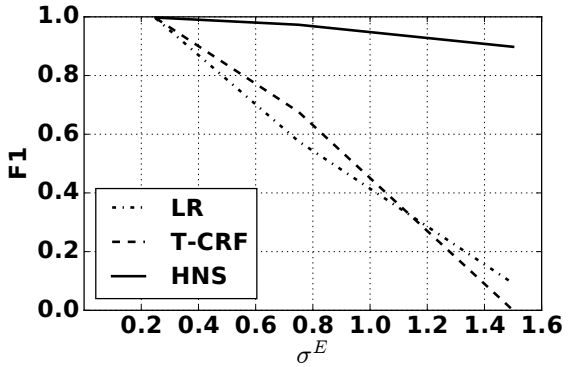
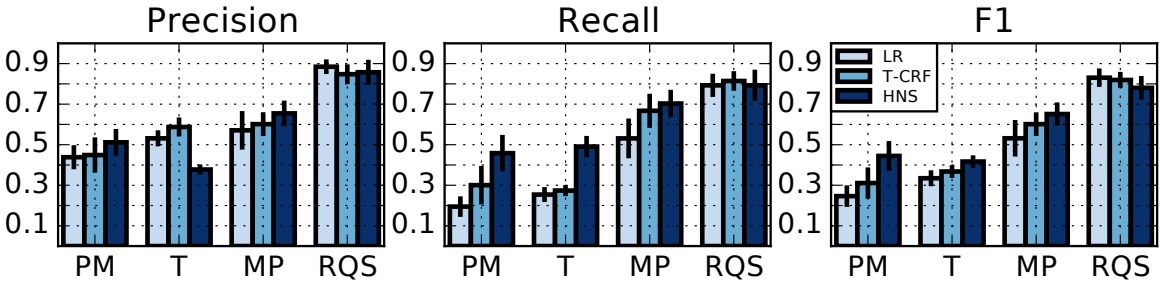


Figure 4. F_1 results for the LR, T-CRF, and HNS models on synthetic data for structural standard deviation $\sigma^S = 0.25$ with varying amounts of event noise σ^E.

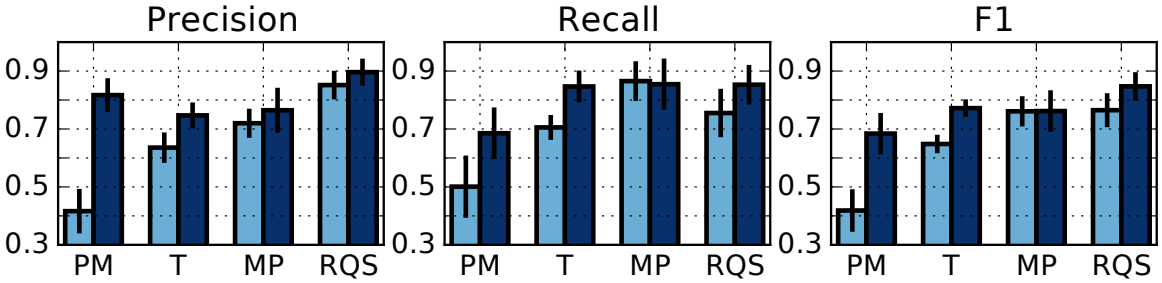
Synthetic Experiments and Results: To evaluate the performance of the HNS model under controled noise conditions, we evaluated all models on a series of synthetic datasets. For each synthetic data case, we sampled the length of a session, the number of positive events per session, and the number of negative events between positive events from discretized, truncated normal distributions with standard deviation $\sigma^S = 0.25$. Next, we sampled event-level features from class conditional normal distributions with means separated by unit distance, and a common standard deviation parameter σ^E, which we varied to simulate different amounts of discriminative information. We generated train, validation, and test sets containing 30, 50, and 50 data cases of length 100 respectively.

Figure 4 shows the event level F_1 score for each model versus the event standard deviation (σ^E). When there is little noise in the features ($\sigma^E = 0.25$), all methods perform equally well; however, the HNS model substantially outperforms the other two models when there is there is less information in the event features, indicating that the HNS model can more effectively leverage higher level structure in the data.

Real Data Results: The results from our Task 1 event detection experiments on the mHealth benchmark datasets are shown in Figure 3a. The HNS model performs better in terms of average F_1 score on three of the four data sets. On



(a) Task 1: Event Detection



(b) Task 2: Activity Segmentation

Figure 3. The top row shows Task 1 (event detection) results while the second row shows Task 2 (activity segmentation) results. From left to right, the three panels in each row correspond to precision, recall, and F1. In each group of bars for Task 1, the models are LR, T-CRF, and HNS. In each group of bars for Task 2, the models are T-CRF, and HNS.

the T dataset, these results are statistically significant at the $p = 0.05$ level using Bonferroni correction. In addition, we ran a paired t-test on the combined results of all datasets and found that the HNS model achieves an improvement over both LR and T-CRF that is statistically significant at the $p = 0.05$ level, again using Bonferroni correction.

The results from the Task 2 segmentation experiments are shown in Figure 3b. The HNS model outperforms the T-CRF baseline in terms of F_1 score on three of the four datasets and has the same performance on the MP dataset. The improvements range from 0.082 to 0.266 F1 and are statistically significant at the $p = 0.05$ level on the PM and T datasets. When all datasets are considered together, the improvement in segmentation F_1 over the T-CRF model is significant at the $p = 0.05$ level. Finally, we note that we ran the same set of experiments using maximum likelihood learning for the HNS model; however, segmentation performance was uniformly worse across all datasets.

Unfortunately, our results on Task 2 are not easily comparable to the original papers in which the data sets appeared as these papers do not consider the segmentation problem directly. Additionally, on the event labeling task, Ali et al. (2012) evaluate on rebalanced data, Saleheen et al. (2015) use ad-hoc pre-filtering methods to form train and test sets, and the exact data used for evaluation in Parate et al. (2014) is not available. Our implementation of the random forest experiment from Thomaz et al. (2015) achieves Task 1 F_1 score of 0.31[2]. This is very close to the performance of LR on the same task indicating that performance is not limited by the choice of a linear model, at least on the T dataset.

5. Conclusions and Future Work

In this paper, we have addressed the problem of nested hierarchical segmentation and labeling of event sequences derived from wireless on-body sensor data streams. The primary contributions of the paper are the proposal of a novel model and an efficient MAP inference algorithm for solving both tasks jointly. We have shown that the proposed model significantly out-performs a strong baseline consisting of a pairwise tree-structured CRF designed specifically for two-level segmentation.

In terms of future work, we note that further improvements can likely be derived from better engineered features (or feature learning), as well as learning per-subject models to deal with between subject variability. More broadly, this work can be combined with other context variables and detectors for cognitive states like stress to begin modeling the relationships between cognitive state and negative health behaviors. Finally, there is significant interest in moving this work to the real-time setting to enable continuous health and behavior monitoring applications.

[2]This number differs somewhat from the performance reported in (Thomaz et al., 2015) due to differences in the train/test splits and the way results were averaged.

Acknowledgments

The authors would like to thank Deepak Ganesan and Gregory Abowd for helpful discussions and support of this research. This work was partially supported by the National Institutes of Health under awards R01DA033733, R01DA035502, 1R01CA190329, R01MD010362, and 1U54EB020404, and the National Science Foundation under awards IIS-1350522 and IIS-1231754.

References

Ali, Amin Ahsan, Hossain, Syed Monowar, Hovsepian, Karen, Rahman, Md Mahbubur, Plarre, Kurt, and Kumar, Santosh. mPuff: Automated Detection of Cigarette Smoking Puffs From Respiration Measurements. In *Proceedings of the 11th International Conference on Information Processing in Sensor Networks*, pp. 269–280. ACM, 2012.

DeVol, Ross, Bedroussian, Armen, Charuworn, Anita, Chatterjee, Anusuya, Kim, In Kyu, Kim, Soojung, and Klowden, Kevin. An unhealthy america: The economic burden of chronic disease. 2007.

Finkel, Jenny Rose, Kleeman, Alex, and Manning, Christopher D. Efficient, Feature-based, Conditional Random Field Parsing. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, volume 46, pp. 959–967, 2008.

Kschischang, Frank R, Frey, Brendan J, and Loeliger, H-A. Factor Graphs and the Sum-Product Algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, 2001.

Kumar, Santosh, Nilsen, Wendy, Pavel, Misha, and Srivastava, Mani. Mobile Health: Revolutionizing Healthcare Through Transdisciplinary Research. *Computer*, (1):28–35, 2013.

Lafferty, John, McCallum, Andrew, and Pereira, Fernando C N. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. 2001.

McGinnis, J Michael, Williams-Russo, Pamela, and Knickman, James R. The case for more active policy attention to health promotion. *Health affairs*, 21(2):78–93, 2002.

Mokdad, Ali H, Marks, James S, Stroup, Donna F, and Gerberding, Julie L. Actual causes of death in the united states, 2000. *Jama*, 291(10):1238–1245, 2004.

Müller, Andreas C and Behnke, Sven. Pystruct: Learning structured prediction in python. *The Journal of Machine Learning Research*, 15(1):2055–2060, 2014.

Parate, Abhinav, Chiu, Meng-Chieh, Chadowitz, Chaniel, Ganesan, Deepak, and Kalogerakis, Evangelos. RisQ: recognizing smoking gestures with inertial sensors on a wristband. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*, pp. 149–161. ACM, 2014.

Plath, Nils, Toussaint, Marc, and Nakajima, Shinichi. Multi-class image segmentation using conditional random fields and global classification. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 817–824. ACM, 2009.

Reynolds, Jordan and Murphy, Kevin. Figure-ground segmentation using a hierarchical conditional random field. In *Computer and Robot Vision*, pp. 175–182. IEEE, 2007.

Saleheen, Nazir, Ali, Amin Ahsan, Hossain, Syed Monowar, Sarker, Hillol, Chatterjee, Soujanya, Marlin, Benjamin, Ertin, Emre, Al'Absi, Mustafa, and Kumar, Santosh. puffMarker: A Multi-sensor Approach for Pinpointing the Timing of First Lapse in Smoking Cessation. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 999–1010, 2015.

Sarawagi, Sunita and Cohen, William W. Semi-markov conditional random fields for information extraction. In *Advances in Neural Information Processing Systems*, pp. 1185–1192, 2004.

Shiffman, Saul, Stone, Arthur A, and Hufford, Michael R. Ecological momentary assessment. *Annual Review of Clinical Psychology*, 4:1–32, 2008.

Shotton, Jamie, Winn, John, Rother, Carsten, and Criminisi, Antonio. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *Proceedings of the European Conference on Computer Vision*, pp. 1–15. Springer, 2006.

Smith, David A and Eisner, Jason. Dependency parsing by belief propagation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 145–156. Association for Computational Linguistics, 2008.

Sung, Jaeyong, Ponce, Colin, Selman, Bart, and Saxena, Ashutosh. Unstructured human activity detection from rgbd images. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 842–849. IEEE, 2012.

Sutton, Charles and McCallum, Andrew. An introduction to conditional random fields for relational learning. *Introduction to statistical relational learning*, pp. 93–128, 2006.

Tang, Kevin, Fei-Fei, Li, and Koller, Daphne. Learning latent temporal structure for complex event detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1250–1257. IEEE, 2012.

Taskar, Ben, Guestrin, Carlos, and Koller, Daphne. Max-margin Markov networks. *Advances in neural information processing systems*, 16:25, 2004a.

Taskar, Ben, Klein, Dan, Collins, Michael, Koller, Daphne, and Manning, Christopher D. Max-Margin Parsing. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, volume 1, pp. 3. Citeseer, 2004b.

Thomaz, Edison, Essa, Irfan, and Abowd, Gregory D. A Practical Approach for Recognizing Eating Moments with Wrist-mounted Inertial Sensing. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '15, pp. 1029–1040. ACM, 2015.

Tsochantaridis, Ioannis, Joachims, Thorsten, Hofmann, Thomas, and Altun, Yasemin. Large margin methods for structured and interdependent output variables. In *Journal of Machine Learning Research*, pp. 1453–1484, 2005.

Verbeek, Jakob and Triggs, William. Scene segmentation with CRFs learned from partially labeled images. 20: 1553–1560, 2008.