# Robust indoor activity recognition via boosting

Masamichi Shimosaka, Taketoshi Mori and Tomomasa Sato
Department of Mechano-Informatics, the University of Tokyo, Japan
{simosaka,tmori}@ics.t.u-tokyo.ac.jp, tomomasasato@jcom.home.ne.jp

## Abstract

*In this paper, a novel statistical indoor activity recognition algorithm is introduced. While conditional random fields (CRFs) have prominent properties to this task, no optimal performance is obtained due to the fact that the performance is optimized for offline estimation. Furthermore, no previous researches provide efficient training process to optimize classifiers in on-site recognition perspective. In this paper, we propose a novel sequence estimation model suitable for online activity recognition, what we call Just-in-Time random fields (JRFs). In JRFs, efficient training and feature selection process is provided via boosting. Empirical evaluation using synthetic and real indoor activity records shows that our model drastically outperforms the previous methods in view of the classification performance with respect to the training cost.*

## 1. Introduction

Recent years, rapid growth of digital technologies promotes researches on "life log (lifeLog)" [1]. Activity recognition, which annotates (segmenting and labeling) sensor records with human activity tag, is one of essential foundations to innovate practical applications on massive lifeLog and to support smoothly human daily life. For pattern recognition research, it is desirable to formulate activity recognition as a statistical sequence labeling problem where the output is a sequence of labels rather than a single label, as well as POS tagging in natural language processing and speech recognition.

To achieve robust sequence labeling, it is important to leverage Markov property of human activity because human requires a certain time interval to behave some activity. Among many statistical models for sequence labeling, conditional random field (CRF) [2], a specific model of Markov random fields for sequence labeling, has been applied successfully to many annotation tasks [3, 4]. Compared to hidden Markov models (HMMs), CRFs tend to gain superior performance due to the fact that CRFs can incorporate over-lapping features or prior knowledge of the domain. CRFs will be powerful tools for activity recognition, however, the fatal drawback of CRFs arises in *on-site* or *online* recognition task. This is because they are assumed to be used for *offline* or *batch* activity recognition and they adjust the parameters so as to be optimal for offline activity recognition.

In this paper, we propose an alternative CRF-like discriminative sequence labeling framework that resolves the above problem of CRFs, what we call Just-in-Time random fields (JRFs), so as to be optimal in online activity recognition. It is well known problem that the most training algorithms for CRF-like models [5, 6, 7] require brute force computation to complete the learning process. Hence we innovate efficient learning scheme for JRFs to make them feasible. As well as efficiency for training cost, feature selection is also needed because feature selection makes classifiers compact and readable. To realize feature selection for CRF-like estimators, one may utilize kinds of axis ascent techniques [8] or regularized training with Laplace prior [7]. The former cannot guarantee the global optimality, and the additional computational cost is needed in the latter case. Thus, practical training schemes for JRFs, which provide both parameters estimation and feature selection efficiently, must be investigated. In this paper, we propose a novel framework based on boosting [9]. Our method achieves high optimality with much smaller computational cost than the previous approaches.

## 2. Desirable property of activity recognition

### 2.1. Problem setting of activity recognition

Before formulating activity recognition as statistical sequence labeling, some variables and notations are described. Sensor data are depicted by $\boldsymbol{x}$ and the sequence of the sensor data from time 1 to $t$ is represented by $\boldsymbol{x}_{1:t}$. The corresponding human activity is depicted by $y_t$, a discrete variable representing such activities as sleeping, meal and watching TV. In activity recognition, $\boldsymbol{x}_t$ contains sensor information acquired by sensors embedded in rooms. Even though sensor information $\boldsymbol{x}_t$ provides cues to classify the corresponding activity la-

bel $y_t$, it is natural to incorporate consecutive activity label so as to improve classification performance. This is because watching TV would continue at the next moment. On the other hand, sleeping never occurs at the next moment of meals. Hence, we tackle activity recognition as sequence labeling problem that considers association of the consecutive activity label of $y_{1:t}$.

### 2.2. Problem of conditional random fields

In general, sequence labelers focus on the combination of the whole sequence label $y_{1:T}$ from the sensor data $\boldsymbol{x}_{1:T}$ rather than single label estimation $y_t$ from $\boldsymbol{x}_t$. CRFs are designed to be optimal to solve this problem. In CRFs, $p(y_{1:T}|\boldsymbol{x}_{1:T})$ is defined and optimized so as to provide maximum log likelihood to the pair $\boldsymbol{x}_{1:T}$ and $y_{1:T}$. In inference, CRFs output $\hat{y}_{1:T}$ that satisfies maximum log likelihood to given input sequence $\boldsymbol{x}_{1:T}$. Viterbi algorithm [7] is an efficient and exact inference to decode human activity, however, $\boldsymbol{x}_{t+1:T}$ and $y_{t+1:T}$ is needed to calculate the optimal posterior probability $p(y_t|\boldsymbol{x}_{1:T})$. For on-site activity estimation, the global optimal decoding in CRFs is infeasible.

### 2.3. Just-in-Time classification

To tackle the above problem of CRFs, we must investigate optimal online sequence classification and training framework. In this context, *online* means that activity must be inferred from the current and past sensor information, and the past activity recognition result. In this research, the online optimal classification framework is called Just-in-Time classification. Though HMMs and CRFs can be used in online classification, the optimality of the posterior $p(y_t|\boldsymbol{x}_{1:t})$ is not focused in the training process. Hence we must directly optimize the posterior probability $p(y_t|\boldsymbol{x}_{1:T})$ in Just-in-Time classification framework.

### 2.4. Other practical consideration

Feature selection framework would be appreciated to remove redundant sensor information, to avoid overfitting problems, and to provide readability the model from the parameter information. Note that it is difficult to realize efficient feature selection in HMMs and CRFs. In HMMs, one needs laborious work to select discriminative features before training the model. As well as HMMs, feature selection is laborious in case of training CRFs with Gaussian prior [7]. To realize systematic feature selection, one can leverage Laplace prior [7], but the computational cost drastically grows in the large dataset. Thus, we must design model where efficient feature selection is available.

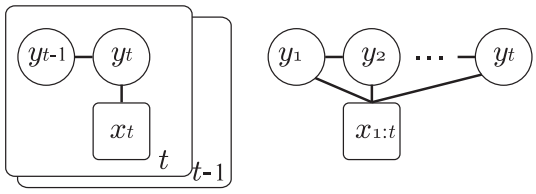


**Figure 1. Graphical models of JRFs (Left) and CRFs (Right) are shown.**

## 3. Just-in-time random fields

### 3.1. Model representation and the inference

Just-in-Time classification framework focuses on the posterior probability $p(y_t|\boldsymbol{x}_{1:t})$ successively. Hence, a model used as Just-in-Time classifier needs recursive computation of the posterior probability $p(y_t|\boldsymbol{x}_{1:t})$. We formulate the posterior by the following equation,

$$p(y_t|\boldsymbol{x}_{1:t}) \propto \exp\left(F_{y_t}(\boldsymbol{x}_{1:t}) + \langle G_{y_t}(y_{t-1})\rangle_{\tilde{p}(y_{t-1})}\right), \quad (1)$$

where $\tilde{p}(y_{t-1}) = p(y_{t-1}|\boldsymbol{x}_{1:t-1})$ denotes the posterior of the last moment, and $\langle f\rangle_q$ depicts the expected score of $f$ over distribution $q$. In (1), discriminative functions $F(\cdot)$ and $G(\cdot)$ are obtained through learning process. The former denotes the discriminative function depending on the current or past sensor information. The latter is calculated by using the last activity information. The probability of label $c$ at time $t$ depends on $\boldsymbol{x}_{1:t}$, and the transition probability of the current and the last activity estimation. Figure 1 shows that the graphical representation of JRFs. The model is similar to maximum entropy Markov models (MEMMs) [10], however, our model avoids a well known label-bias problem [2]. MEMM formulates the maximum entropy distribution over $y_t$ at each last label $y_{t-1}$, thus, the information $y_{t-1}$ itself is not inserted into the distribution. On the other hand, our model contains $y_{t-1}$ information directly to the probability of $y_t$.

### 3.2. Learning JRFs via boosting

Following from (1), the discriminative functions $F(\cdot)$ and $G(\cdot)$ are determined during the learning process. In this research, we use the following maximum likelihood criterion,

$$\hat{F}(\cdot), \hat{G}(\cdot) = \underset{F,G}{\operatorname{argmax}} \sum_t \ln p(y_t|\boldsymbol{x}_{1:t}). \quad (2)$$

To obtain the optimal functions, we adopt the idea of functional gradient approach of boosting [9] to optimize functions $F, G$ iteratively. Specifically, the algorithm starts the function with neutral $F \equiv 0$, $G \equiv 0$, then sequentially updates the function as $F \leftarrow F + \tilde{f}$ and

$G \leftarrow G + \tilde{g}$ to provide better likelihood score. In each iteration, the functions $\tilde{f}, \tilde{g}$ are calculated via weighted least square (WLS) problem. The WLS problem is summarized as follows.

$$\tilde{f}_c \text{ or } \tilde{g}_c = \underset{\phi}{\operatorname{argmin}} \sum_t w_{t,c} \left(\phi - z_{t,c}\right)^2 \quad (3)$$

$$z_{t,c} = \begin{cases} 1/\tilde{p}(y_t = c) & \text{if } y_t = c \\ -1/(1 - \tilde{p}(y_t = c)) & \text{if } y_t \neq c \end{cases}, \quad (4)$$

where $w_{t,c}$ represents $\tilde{p}(y_t = c)(1 - \tilde{p}(y_t = c))$. The training process iterates to calculate the posterior probability $\tilde{p}(y_t)$ and to update functions $F, G$. One can find that this training process is similar to the expectation and maximization algorithm [7]. In view of boosting framework, functions in each iteration $\tilde{f}$, $\tilde{g}$ are called weak classifiers or learners. Note that the training process only depends on the number of the iterations. This is a great property in view of practitioners.

### 3.3. Feature selectable weak classifiers

An important factor to make our algorithm practical is to design good and simple weak classifiers. Note that a certain kind of combination of features in solving the above WLS problem, the algorithm is essentially performing feature selection and parameter estimation.

For determining discriminative sensors, we leverage decision stump to formulate $\tilde{f}$ because stumps provide both systematic feature selection and readability of the model. In this research, decision stump $\tilde{f}_c(\boldsymbol{x}_t)$ outputs two discrete values from $q$-th attributes of $\boldsymbol{x}_t$ as

$$\tilde{f}_c(\boldsymbol{x}_t) = \begin{cases} a_c & \text{if } \{\boldsymbol{x}_t\}_q > \theta \\ a'_c & \text{if } \{\boldsymbol{x}_t\}_q \leq \theta \end{cases}. \quad (5)$$

The optimal parameters $a_c$, $a'_c$ can be obtained as

$$a_c = \frac{\sum_t w_{t,c} z_{t,c} [\![\{\boldsymbol{x}_t\}_q > \theta]\!]}{\sum_t w_{t,c} [\![\{\boldsymbol{x}_t\}_q > \theta]\!]} \quad (6)$$

$$a'_c = \frac{\sum_t w_{t,c} z_{t,c} [\![\{\boldsymbol{x}_t\}_q \leq \theta]\!]}{\sum_t w_{t,c} [\![\{\boldsymbol{x}_t\}_q \leq \theta]\!]}, \quad (7)$$

when threshold $\theta$ is given. The optimal $\theta$ is easily obtained via simple bounded line search optimization method. $\tilde{g}$ differs from $\tilde{f}$ in that the input is discrete. In this paper, we model the weak classifier $\tilde{g}$ as $\tilde{g}_c(y_{t-1}) = a_{y_{t-1}}$. The above WLS problem provides optimal solution of $a_c$ as

$$a_{c'} = \frac{\sum_t w_{t,c} z_{t,c} [\![y_{t-1} = c']\!]}{\sum_t w_{t,c} [\![y_{t-1} = c']\!]}. \quad (8)$$

## 4. Experimental results

We validate the performance of JRFs and compare them with other labeling methods. We execute experiments using both synthetic and real activity data.

### 4.1. Classification task with synthetic dataset

**Dataset:** The synthetic dataset is generated by using a first order hidden Markov model. The number of the states and the dimensionality of the model is 3 and 3, respectively. The length of the generated sequence is around 100 to 140. The dataset contains 40 sequences. The model cyclically translates in the three states where the probability to stay the same symbol during the consecutive frames is 90%. We use single Gaussian distribution for each state emitter.

**Evaluation method:** To validate our superiority, we compare it with CRF online filtering process, and multiclass version of LogitBoost [9]. The decoding process of CRF filtering process is calculated via forward sum product algorithm [2]. This method does not execute backtrack procedure as in Viterbi decoding. Each method can estimate on-site label $y_t$ from the given sequence $\boldsymbol{x}_{1:t}$. As a reference to calculate the best performance for sequence labeling (including offline case), we utilized CRFs with Viterbi algorithm. The evaluation criteria used here are accuracy and computational cost. We calculate interval to complete learning process as the latter criterion. We utilized cross validation technique to calculate reliable performance score.

**Parameters and condition:** In JRFs and LogitBoost, we determined the number of the iteration as $40$. As for CRF training, we incorporate regularization factor with Gaussian prior. Specifically, the process maximizes $\ln p(y_{1:T}|\boldsymbol{x}_{1:T}) - C\boldsymbol{w}^{\mathsf{T}}\boldsymbol{w}$. We set $C = 0.25$ to provide the best classification rate. In the experiment we utilized 20 sequences as training data and the rest for evaluation. The performance is obtained through 10 times simulation for statistical reliability. L-BFGS quasi-Newton algorithm is used in CRF training. We utilize a modern laptop computer with Intel Core 2 Duo CPU and 1.5GB RAM.

**Result:** The result of this experiment is shown in Table 1. B-JRF represents *JRFs via boosting*. This result implies that JRFs provide the best performance. As a reference score, Viterbi CRFs provide $95.5 \pm 0.5$ %. This is better performance than JRFs, however, this case is permitted only when offline classification is available. Because JRFs directly optimize online classification performance, they outperform CRFs filter. Furthermore, our training process requires much less computational cost than the case of CRFs. The effectiveness to incorporate the function $\tilde{g}$ is appreciated because the LogitBoost classifier lacks $\tilde{g}$.

**Table 1. Results on synthetic dataset**

| | Accuracy (%) | Time (sec.) |
|---|---|---|
| B-JRF | $89.9 \pm 1.0$ | $1.2 \pm 0.2$ |
| CRF | $82.9 \pm 3.1$ | $25.0 \pm 4.3$ |
| LogitBoost | $77.9 \pm 0.7$ | $1.2 \pm 0.2$ |

### 4.2. Real indoor activity recognition

**Dataset:** To capture massive sensor data, we utilized *Sensing Room* [11], an implementation of room-typed sensor accumulation space. They contain pressure sensors of the floor and switching sensors in order to detect that one touches furniture. The system measures sensor data at 1 Hertz. They capture 8 sequences with about 1500 minutes in total, which are recorded through 1 week period. In this experiment, we design activity label so that we can survey one day activity at a glance. Specifically, we design the activity with 10 categories: *Going out*, *Moving inside the room*, *(Playing) Game*, *Watching TV*, *Meal*, *Meal preparation*, *Washing (hands or faces)*, *Deskwork*, *Rest in sofa (Relax)*, *Sleeping*. Note that the label *Game*, *Watching TV*, *Meal* represents activities around the chair at the dining table, and *Meal preparation* and *Washing* occur when an resident stay at sink cabinet. Thus, we must consider contextual information of the label to realize robust activity recognition. We segmented the sequence into 10 seconds chunks and manually labeled activity tag at each chunk. For each window of the sequence, we leveraged 424 sensors information. The values at each chunk are averaged by 10 times measurement in the same chunk.

**Evaluation method:** We evaluate the performance of JRFs and compare it with LogitBoost, (Viterbi) CRFs. In this experiment, we employ F-measure as a performance criterion. This value is harmonic mean of precision and recall. F-measure $\mathcal{F}$ can be defined as $\mathcal{F} = \frac{2\mathcal{P}\mathcal{R}}{\mathcal{P}+\mathcal{R}}$, where $\mathcal{R}, \mathcal{P}$ denotes recall, precision, respectively. $\mathcal{P}$ represents accuracy to the detection frames and $\mathcal{R}$ represents accuracy to detection performance. F measure $\mathcal{F}$, which ranges from 0 to 1, gains higher value when the classifier provides good performance. We obtained $\mathcal{F}$ through the leave-one-out cross validation technique [7]. We empirically determined the regularization parameter of the Gaussian prior of CRFs and the number of the iteration of JRFs and LogitBoost.

**Result:** Table 2 shows that F-measure of each method for real activity annotation. Our method achieves completive classification performance to that of (offline optimal) Viterbi CRFs. The training cost of JRFs requires only 3 hours learning process, whereas over 24 hours computation is required in Viterbi CRFs. Note that the performance drastically decreased in CRFs with forward sum product algorithms. In LogitBoost, the performance of classifying *Move*, and *Meal preparation* drastically descended. This implies Markov assumption is invaluable for activity recognition. The experimental results indicate that our model provides excellent property in view of classification performance per the training cost.

**Table 2. Results on real activity records**

| Label | B-JRF | (Viterbi) CRF | LogitBoost |
|---|---|---|---|
| 1 Out | 82.9% | 92.0% | 56.4% |
| 2 Move | 59.2% | 59.4% | 28.8% |
| 3 Game | 99.7% | 100.0% | 99.8% |
| 4 Watching TV | 88.4% | 90.3% | 91.1% |
| 5 Meal | 51.4% | 50.1% | 46.0% |
| 6 Meal prep. | 56.5% | 68.0% | 47.7% |
| 7 Washing | 68.0% | 37.0% | 60.6% |
| 8 Deskwork | 97.9% | 99.7% | 98.3% |
| 9 Relax. | 99.8% | 99.3% | 99.7% |
| 10 Sleep | 99.1% | 92.7% | 98.9% |

## 5. Conclusion

In this paper, we propose a novel indoor activity recognition algorithm. We design the algorithm, what we call Just-in-Time random fields (JRFs), so as to be optimal to classify human activity in online situation. In this paper, we also introduced an efficient simultaneous feature selection and parameter optimization via boosting. Empirical evaluation using synthetic dataset and real activity dataset shows that our model provides drastically outperforms the previous methods in view of the classification performance with respect to the training cost. In future work, we plan to build sophisticated multi-label (tagging) problem in activity recognition.

## References


[1] R. Lueder S. Drucker J. Gemmell, G. Bell and C. Wong. MyLifeBits: fulfilling the memex vision. *ACM Multimedia System Journal*, pages 235–238, 2002.

[2] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In *Proc. of the 18th ICML*, pages 282–289, 2001.

[3] S. Kumar and M. Hebert. Discriminative random fields: a discriminative framework for contextual interaction in classification. In *Proc. of 9th ICCV*, volume 2, pages 1150–1157, 2003.

[4] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Conditional random fields for contextual human motion recognition. In *Proc. of the 10th ICCV*, volume 2, pages 1808–1815, 2005.

[5] Y. Altun, M. Johnson, and T. Hofmann. Investigating loss functions and optimization methods for discriminative learning of label sequences . In *Proc. of EMNLP 2003*.

[6] A. Torralba, K. Murphy, and W. Freeman. Contextual models for object detection using boosted random fields. In *Advances in NIPS 17*, pages 1401–1408, 2005.

[7] D. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[8] A. McCallum. Efficiently inducing features of conditional random fields. In *Proc. of 9th UAI*, pages 403–410, 2003.

[9] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. Technical report, Department of Statistics, Stanford University, 1998.

[10] A. McCallum, D. Freitag, and F. Pereira. Maximum entropy Markov models for information extraction and segmentation. In *Proc. of the 17th ICML*, pages 591–598, 2000.

[11] H. Noguchi, T. Mori, and T. Sato. Construction of data accumulation system for human behavior information in room. In *Proc. of IROS 2002*, pages 1252–1258, 2002.