







Uniform Evaluation of Properties in Activity Recognition

Seyed M. R. Modaresi^{1,2}(✉) , Aomar Osmani² ,
Mohammadreza Razzazi^{1,4} , and Abdelghani Chibani³ 

¹ SSRD Lab., Computer Engineering Department, Amirkabir University
of Technology, Tehran, Iran

razzazi@aut.ac.ir

² LIPN-UMR-CNRS 7030 Lab., Sorbonne University Paris Nord, Paris, France

{modaresi,aomar.osmani}@lipn.univ-paris13.fr

³ LISSI Lab., Université Paris-Est Créteil, Paris, France

chibani@u-pec.fr

⁴ Computer Science, Institute for Research in Fundamental Sciences, Tehran, Iran

Abstract. The main additional problem in activity recognition (AR) systems in contrast to traditional ones is the importance of duration: a predicted concept in AR is durative and can be correct in a period and incorrect in another one. Therefore, it is fundamental to extend the correctness vocabulary and to formalize a new evaluation system including these extensions. Even in similar areas, few empirical attempts are proposed which are confronted with the problems of correctness and completeness. In this paper, we propose the first formal multi-modal evaluation approach for durative concepts. This novel mathematical method evaluates the performance of an AR system from multiple perspectives, including detection, total duration, relative duration, boundary alignment, and uniformity. It extracts the properties considered in the state-of-the-art and redefines the well-known true-positive, false-positive and false-negative terms for durative events. Our proposed method is extensible, interpretable, customizable, open source and improves the expressiveness of the evaluation while its computation complexity remains linear. Comprehensive experimental evaluations are conducted to show the usefulness of our proposed method.

Keywords: Evaluation · Activity recognition · Time series

1 Introduction

Activity Recognition (AR) is expected to be a core component in numerous future Internet of Things applications such as healthcare, smart homes, and security [5, 22, 23]. Therefore, evaluating the effectiveness of different AR algorithms is essential. Some metrics such as accuracy, observing the recall against precision are common metrics that are easy to understand and interpret even by non-experts. These metrics are well-used for discrete instances and pre-segmented

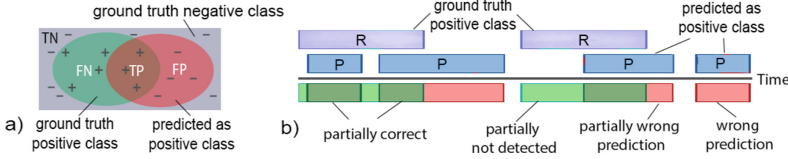


Fig. 1. a) Classical instances b) Durative instances. Durative one may partially correct and partially incorrect while the classical one is either correct or not.

Table 1. The notations used in this paper.

Symbol	Description
TP, TN, FP, FN	True Positive, True Negative, False Positive, False Negative
$e=(c, [s:f])$	Event e has class c that occurs from time s to time f
GTE, PE	GTE=Ground Truth Event, PE = Predicted Event
E, R, P	E = Event set, R = GTE set, P = PE set
$ X $	The number of element in set X
$T(e), T(E)$	$T(e)$ = Duration of e , $T(E) = \sum_{e \in E} T(e)$
$e' = e_1 \cap e_2$	$e_1 = (c_1, t_1) \wedge e_2 = (c_2, t_2) \wedge c_1 == c_2 \wedge e' = (c_1, t_1 \cap t_2)$
$e_1 \cap E_2$	$\bigcup_{e_2 \in E_2} e_1 \cap e_2$
$[\cdot]$	Iverson bracket. 1 when the enclosed condition is true; otherwise, 0

data sequences [5]; where, a predicted instance is either correct or incorrect. However, concepts in AR are durative; thus, a predicted concept can be correct in one period, incorrect or partially correct in another one [26]. Accordingly, as shown in Fig. 1, previously well-defined terms used in traditional systems such as true positive (TP), false positive (FP), false negative (FN) are not suitable for durative concepts [26].

However, it is often assumed that time-frame, event-based, or classifier performance follows the whole system performance [4, 5, 22, 23]. This assumption neglects practical scenarios and may misleadingly present convincing results (Sect. 2). Despite the importance of evaluating durative concepts, it is not well-developed even in other areas. Still, there is no universally accepted formula for evaluating the effectiveness of systems with durative concepts.

This paper proposes a novel mathematical method for evaluating different properties of AR systems. It redefines TP, FP, and FN to consider various properties such as detection, total duration, relative duration, boundary alignment, and uniformity between ground truth and predicted events. Therefore, confusion matrix based metrics such as recall, precision, and f-score, can be calculated to evaluate and compare different systems. Furthermore, it is simple, time-efficient, extensible, and customizable. It also overcomes the limitations of existing methods. Although, it can select an appropriate algorithm for a new application by prioritizing properties differently. The experiments show that our method can outperform state-of-the-art methods with enhanced generalization capability.

2 Preliminaries and Related Work

Evaluating the performance of AR systems is usually done by comparing predicted events (PEs) with the ground-truth events (GTEs) [16]. It can be viewed as the matching of two time-series. However, it is not easy to determine the time boundaries of ground truth labels perfectly; moreover, the distinction between activities is not always clear [5]. Therefore, some decision functions accommodate offsets using *ambiguous range* [10], *fuzzy event boundaries* [20], time series matching techniques (such as *dynamic time warping*, *longest common sub-sequences* [7]), or *categorical probability distribution* [9]; however, they fail to distinguish different types of errors (e.g., fragmentation) [27]. Common approaches to evaluate AR systems include time-frame, event-based, and classifier performance [12, 17, 23]. *Time-frame based* methods uses fixed period interval as atomic units and facilitate comparing different AR algorithms since each frame is independent of both the GTEs and PEs [12, 17]. Nevertheless, the interpretation of errors is not the same in different applications. Hence, each frame’s error is classified to *insertion* (detection of an activity when nothing actually happened), *overflow* (time before and after the occurrence time of an activity that is incorrectly identified as part of the activity), and *merge* (covering multiple GTEs by a single PE) as sources of FP errors and *deletion* (failure to detect an activity), *substitutions* (wrongly detected with another class), *underfill* (not detected duration at the beginning and end of the activity), and *fragmentation* (detecting a GTE by multiple PEs) as sources of FN errors [17]. Moreover, event based methods are also essential to be considered as well as time-frame [27]. Event based errors are categorized as insertion, deletion, fragmentation, merge and fragmented-merge (occurrence of both merge and fragmentation errors) [27]. However, an expert must do a time-consuming analysis of these massive and heterogeneous diagrams, matrices, and information. Therefore, combining them as a scalar metric is complex. Besides, These approaches also consider the total duration of positional errors and do not provide an event-based tunable model for it.

From the behavior analysis perspective, evaluating each activity needs a different evaluation method [1]. e.g., duration sensitive activities need to be evaluated differently from frequency sensitive ones. Timeliness is another metric used for online and realtime prediction [24]. It is defined as the duration continuous correct prediction of an activity without switching to an inaccurate prediction. To compare different AR algorithms in a similar situation, a competition is held and time frame f_1 -score, recognition delay, installation complexity, user acceptance, and interoperability are used as the evaluation criteria [8].

In sound event detection (SED) [4], video action detection [3], anomaly detection [26], and video abnormal event detection [11], etc., concepts are also durable. The IEEE Audio and Acoustic Signal Processing challenge [25] highlights the need for an appropriate metric in SED. Still, researchers mainly used collar, segment (time-frame based), and PSDS (polyphonic sound detection score) methods [4, 16]. However, they can not show the different sources of errors. Our recent work dedicated to multimodal metrics in SED system [18] provides some evaluation approaches depending on the hypothesis and constraints on SED applications. National Institute of Standards and Technology (NIST) developed

a challenge for detecting activities in video (ActEV) [3]. It firstly used false alarms rate (instance based) and missed detections probability (instance based) as evaluation metrics. However, In 2019, it uses time-frame method for calculating false alarm rate [3]. Other metrics in abnormal event detection in video are false rejection rate, equal error rate, decidability index, receiver operating characteristic curves, and area under the Curve [6, 11]. However, equal error rate can be misleading in the anomaly detection setting [15]. Numenta anomaly benchmark [14] is designed to evaluate different anomaly detection algorithms. It uses a scaled sigmoidal scoring function for the relative position of each detection; however, it ignores fragmented predictions. To resolve previously mentioned issues, researchers in [26] redefine precision and recall for time-series (particularly on anomaly detection). They need some functions to be explicitly defined for a given application. Those functions are: γ (to consider fragmented events), δ (to consider the positional relation between PE and GTE), overlap (the rate of the correctly detected events (e.g., $\text{overlap}(x, y, \delta()) = \mathcal{T}(x \cap y) / \mathcal{T}(x)$), and α which is a coefficient. They are formulated in Eq. (1) using notations of Table 1.

$$\begin{aligned} \text{exist}(e, X) &= [e \cap X \neq \emptyset], & \text{score}(e, X) &= \gamma(e, X) \times \sum_{x \in X} \text{overlap}(e, e \cap x, \delta()), & (1) \\ \text{Recall} &= \frac{1}{|R|} \sum_{r \in R} \alpha \times \text{exist}(r, P) + (1 - \alpha) \times \text{score}(r, P), & \text{Precision} &= \frac{1}{|P|} \sum_{p \in P} \text{score}(p, R) \end{aligned}$$

Issues in [26] (Eq. (1)) are analysed deeply in the following:

1. It surprisingly ignores the α (coefficient) in calculating precision. Therefore, it gives inconsistent weights to *overlap* function in calculating recall and precision. Therefore, to prevent misled interpretation, they can not be used as complementary (e.g., in calculating f1 score).
2. Fragmented PEs have significant positive score in precision. e.g., in Fig. 2, the precision of (a) is much higher than (b). Similar situation happens for recall.
3. It normalizes the duration of events to avoid the duration impacts. Briefly, the precision calculation is $\text{avg}_{p \in P}(\frac{TP}{\mathcal{T}(p)})$ and the recall calculation is $\text{avg}_{r \in R}(\frac{TP}{\mathcal{T}(r)})$.

This normalization looks well for a single PE and GTE; however, in total, it gives different values for TP in recall and precision. Therefore, they are not calculated in a similar mathematical model and they can not be used as complementary (e.g., for f1-score). Equation (2) presents these calculations for Fig. 2 (d).

$$\begin{aligned} \text{Precision} &= \frac{\frac{TP_1}{P_1} + \frac{TP_2}{P_2}}{1 + 1} = \frac{\Sigma \text{normalized TPs based on PEs}}{\Sigma \text{normalized PEs}} & (2) \\ \text{Recall} &= \frac{\frac{TP_1}{R_1} + \frac{TP_2}{R_2} + \frac{0}{R_3}}{1 + 1 + 1} = \frac{\Sigma \text{normalized TPs based on GTEs}}{\Sigma \text{normalized GTEs}} \end{aligned}$$

4. Defining an appropriate cardinality function is complex. Furthermore, it is difficult to adjust and tune this formula since the dependencies between cardinality, position, and overlap are not clear [10]. e.g., in Fig. 2 (c), the first

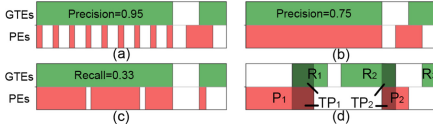


Fig. 2. Example activities that help to explain the drawbacks in [26].

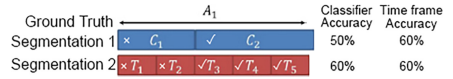


Fig. 3. Evaluation of AR systems that use different segmentation approach.

and second GTEs have the same recall (0.33) (using $\gamma(e, X) = |e \cap X|^{-1}$ as suggested by authors). It is similar for calculating precision for merged PEs.

5. This approach can not be applied to duration-sensitive activities[1].
6. Adding a new property (e.g., total duration) is not straightforward.

Issue in classifier metrics is the inability to compare algorithms in a unified space since AR systems may use various segmentation (windowing) algorithms. Figure 3 is an illustration of two algorithms. Activity A_1 is not detected in segments C_1 , T_1 and T_2 . Thus, the classifier accuracy in the first approach is 50% while it is 60% in the second one. Clearly, the difference in their performances are due to the effects of the different segmentation procedures. Accordingly, it may misleadingly present convincing results and it can not capture duration specific properties, although it is widely used in several papers [5, 7, 13, 19, 23]. Time frame accuracy is more consistent metric [12]; however, it can not displays different property of an AR system such as uniformity, detection of each event or the boundary alignment. Additionally, a long event affect the whole result.

As a result, a new metric is needed to better evaluate AR algorithms while paying attention to the peculiarities of the applications and activities.

3 Proposed Metric

An evaluation method should determine the different properties of AR algorithms. We define a measurement (in terms of recall and precision) for each property, and all together constitute our proposed metrics. A weighted combination of them can produce a scalar value, or they can be used collectively as a multi-objective metric. Because of our approach’s modularity, it can be easily extended to include a measurement for a new property. Our metric is based on the following assumptions: 1- R and P are given as input. 2- Times in concepts are durative and specified. 3- The acceptable time shift of PEs to be assumed as detected is within the GTE range. i.e., PEs and GTEs are relevant when they have some overlap. 4- Only a single activity class is exist. For multi-class cases, all classes are evaluated individually as a positive class and the rest as a negative one. This allows using different parameters for each activity class which is an necessary feature for AR [1]. 5- One instance of an activity class occur at a time.

We use ground truths as references in the normalization process because they are independent of predictions of different algorithms. Therefore, we cluster

GTEs and PEs in such a way $C = \{(r, ps) | r \in R \wedge ps = \{p \in P | r \cap p \neq \emptyset\} \wedge ps \neq \emptyset\}$. Orphan PEs are considered as $\bar{C} = \{p \in P | p \cap R = \emptyset\}$.

Each instance in the classical model is either correctly predicted or not (each TP, FP or FN is either 0 or 1). However, in the durative model, a GTE may be partially covered by positive PEs. Therefore, we allow partial value for TP, FP, and FN. In the following, we present the properties which are drawn from state-of-the-art and our formulas for measuring their values.

Detection (D) Property calculates the detection of a GTE even by a small (at least θ [10]) PE (It checks for the existence of overlaps between PEs and GTEs). A GTE is TP if it is detected at least once and is FN if it is not. PEs that don't have any intersection with any GTEs are considered as FP. This property is useful in applications like alarm systems [26].

$$TP^D = \sum_{(r,ps):C} \left[\sum_{Lp:ps} \frac{\mathcal{T}(r \cap p)}{\mathcal{T}(r)} > \theta_{tp} \right], \quad FN^D = \sum_{(r,ps):C} \left[\sum_{Lp:ps} \frac{\mathcal{T}(p) - \mathcal{T}(r \cap p)}{\mathcal{T}(r)} > \theta_{fp} \right] + |\bar{C}| \quad (3)$$

$$FN^D = |R| - TP^D,$$

Therefore, a GTE is considered as TP when at least θ_{tp} fraction of it is correctly identified; otherwise, it will be considered as FN. FP counts not detected PEs ($|\bar{C}|$) plus the PEs which the rate of its wrong prediction part is higher than θ_{fp} .

Uniformity (U) Property considers the detection of GTE by a single PE instead of multiple fragmented ones. e.g., in a taking medicine event, detecting two taking medicine events instead of one shows a disorder; therefore, the duration is not as important as the number of occurrences. Researchers in [26,27] consider uniformity as an essential property; however, they do not formulate it. Event analysis [27] leads us to consider a GTE as a TP if it is identified by only one PE. In this case, all other PEs are considered as FP or FN.

$$TP^U = \sum_{(r,ps):C} [|\{ps \cap R\}| = 1], \quad FN^U = \sum_{(r,ps):C} [|\{ps \cap R\}| > 1], \quad FP^U = |P| - |\bar{C}| - TP^U \quad (4)$$

Thus, the recognized GTEs are considered as TP if each is detected by one PE and that PE does not identify any other GTEs; otherwise, they are considered as FN. Similarly, a PE, that is neither TP nor orphan, is considered as FP.

Total Duration (T) Property is well-known and is similar to time-frame-based methods. It divides the PEs and GTEs by their boundaries; therefore, each frame is either TP, FP, FN, or TN [12].

$$TP^T = \sum_{(r,ps):C} \mathcal{T}(r \cap ps), \quad FN^T = \mathcal{T}(R) - TP^T, \quad FP^T = \mathcal{T}(P) - TP^T \quad (5)$$

Relative Duration (R) Property normalizes the duration of each event individually to lessen the effect of varying durations of events.

$$\begin{aligned} \text{TP}^{\text{R}} &= \sum_{(r,ps):C} \frac{\mathcal{T}(r \cap ps)}{\mathcal{T}(r)}, & \text{FP}^{\text{R}} &= \sum_{(r,ps):C} \min\left(1, \sum_{p:ps} \frac{\mathcal{T}(p) - \mathcal{T}(r \cap p)}{\mathcal{T}(r)}\right), \\ \text{FN}^{\text{R}} &= |C| - \text{TP}^{\text{R}} \end{aligned} \quad (6)$$

Consequently, TP (FN) is the sum of normalized durations of correctly detected (incorrectly undetected) parts of GTEs. The FP calculation is similar; however, FP of each cluster can not exceed 1.

Boundary Alignment (B_t) Property rewards TP when PEs GTE's boundaries precisely match the boundaries of its related PEs; otherwise, it loses some score by FN (underfill error¹), or FP (overfill error (see footnote 1)) [27]. This property concentrates only on the alignment error and is related to the needs considered in [26, 27]. The parameter t specifies the kind of alignment (start (B_s) or end (B_e)).

$$\begin{aligned} \forall t \in \{\text{start}, \text{end}\}: \quad \text{fn}_1(r, ps) &= \text{if } ps \neq \emptyset \text{ then } 1 - e^{-\beta_t \frac{\text{underfill}_t(r, ps)}{\mathcal{T}(r)}} \text{ else } 0 \\ \text{fp}_1(r, ps) &= \text{if } ps \neq \emptyset \text{ then } 1 - e^{-\beta_t \frac{\text{overfill}_t(r, ps)}{\mathcal{T}(r)}} \text{ else } 0 \\ \text{TP}^{\text{B}_t} &= \sum_{(r,ps):C} \max(0, 1 - \text{fp}_1(r, ps) - \text{fn}_1(r, ps)) \\ \text{FN}^{\text{B}_t} &= \sum_{(r,ps):C} \text{fn}_1(r, ps), & \text{FP}^{\text{B}_t} &= \sum_{(r,ps):C} \text{fp}_1(r, ps) \end{aligned} \quad (7)$$

Accordingly, TP of each cluster is justified by the alignment error between predictions and ground truths. In addition, errors increase exponentially (adjustable with β_t) by the distance between the boundaries of PEs and GTEs. Increasing parameter β_t gives more penalties to longer positional errors.

Precision, Recall, and F-Score are calculated using the following known formula using TPs, FPs, and FNs that were defined earlier for each AR properties.

$$\begin{aligned} \forall f \in \{D, T, R, B_s, B_e, U\}: \quad // \text{Abbreviation of properties} & \quad (8) \\ \text{Recall}^f &= \frac{\text{TP}^f}{\text{TP}^f + \text{FN}^f}, & \text{Precision}^f &= \frac{\text{TP}^f}{\text{TP}^f + \text{FP}^f}, & \text{F}_1^f &= 2 \frac{\text{Precision}^f \cdot \text{Recall}^f}{\text{Precision}^f + \text{Recall}^f} \end{aligned}$$

Computation Complexity of the presented formulas is $O(|R| \times |P|)$ because elements of both sets of P and R are iterated. Since each element of R needs only related P; the interval tree helps us to optimize it to $O(|R| \log |R| + |P| \log |P|)$. In the case that P and R are sorted by time, this complexity can be reduced to $O(|R| + |P|)$ by considering the time relationships of P and R.

¹ $\text{overfill}_{\text{start}}(r, ps) = \max(0, \text{start}(r) - \text{start}(ps))$ $\text{underfill}_{\text{start}}(r, ps) = \max(0, \text{start}(ps) - \text{start}(r))$
 $\text{overfill}_{\text{end}}(r, ps) = \max(0, \text{end}(ps) - \text{end}(r))$ $\text{underfill}_{\text{end}}(r, ps) = \max(0, \text{end}(r) - \text{end}(ps))$.

4 Experimental Results

This section presents an experimental study of our metric. The first experiment is done on small visualizable data. The second one compares two algorithms in a real-world dataset. The parameters of each property of our metric are as follows. The θ_{tp}, θ_{fp} are needed to have an appropriate detection property. In this experiment, if a PE has any overlap with GTE ($\theta_{tp} = 0$), we consider it as TP; additionally, if an incorrect part of a PE is longer than the related GTE's duration ($\theta_{fp} = 1$), we consider it as FP. We also use ($\beta_t = 2$) to consider near linear boundary error. The codes and datasets are existed in our repository at <https://github.com/modaresimr/AR-MME-EVAL>.

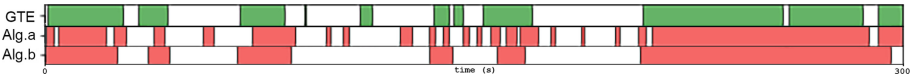
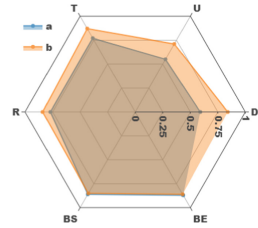


Fig. 4. Ground truths and output of two algorithms used in [27].

Table 2. Details of our metric for algorithms of Fig. 4. The spider chart (right image) shows the f1-score on each property for those algorithms.

Algorithm	Alg.a			Alg.b		
Property	recall	precision	f1	recall	precision	f1
detection	0.73	0.50	0.59	0.73	1.00	0.84
uniformity	0.75	0.43	0.55	0.62	0.83	0.71
total duration	0.78	0.77	0.77	0.84	0.90	0.87
relative duration	0.73	0.81	0.77	0.83	0.85	0.84
boundary start	0.81	0.93	0.86	0.87	0.84	0.85
boundary end	0.99	0.78	0.87	0.85	0.87	0.86



Our Proposed Metric on Small Data is explored in this experiment for simplicity in visualization. This data contains a subset of 13 relations between two intervals in Allen’s interval algebra [21]. This data and our metrics’ outputs are illustrated on Fig. 4 and Table 2. Clearly, more PEs of Alg.a are incorrectly predicted than Alg.b in Fig. 4, while the number of undetected GTEs is the same. The precision and recall in *detection* measurement confirm this observation. The *uniformity* of Alg.b is higher than Alg.a since most of the GTEs detected with a single PE in Alg.b instead of multiple fragmented PEs. For the *total duration* measurement, we can see that the correctly predicted time frames (TP) in Alg.b are more than Alg.a, while it is inverse for the incorrect ones. The *relative duration* normalizes events independently and applies the total duration measurement. It shows Alg.b predict more part each recognized concept than Alg.a. Since the concepts’ duration are similar, the *total duration* shows similar

result. In the *boundary* measurement, we can observe that almost all predictions of Alg.a cover the end boundary of GTEs. Therefore, the end part of all GTEs are well-detected (recall = 0.99); however, there are some part of predictions after end of the GTE’s boundary that are incorrectly predicted (prediction = 0.78).

Our Proposed Metric on a Public Dataset is explored in this experiment. We compare non-overlapping sliding time window of 30 s (SW)² with Hierarchical Hidden Markov model (H-HMM) [2] to show how our metric works. WSU CASAS Home1 dataset [13] that contains 32 sensors, 400,000 events and about 3000 durative concepts (activities) is used in this experiment. We use its first 20% for test and the remaining for training.³ Then we evaluate the effectiveness of *take medicine* activity and the macro average of all classes.⁴ We compare [26] and [27] metrics with ours. The classifier metric issues is discussed in Sect. 2.

Table 5 (b) shows that 50% of times, HHMM algorithm do not detect the concepts and 29% of times it can not detect the start boundary while almost none of its prediction is incorrect. For SW algorithm, it shows great performance except around 16% of times the prediction is fragmented. However, our metric (Table 3) shows this observation is not complete. Analysing the data shows that the duration of 5% of concepts is equal to the others. Therefore, they dominate the system’s quality when using the time frame metrics (e.g., Ward’s time metrics) and classifier metrics⁵. Table 5(a) helps to understand more about the predictions with event analysis perspective. It displays that 28% and 40% of predictions in SW and HHMM algorithms are incorrectly predicted (in contrast to the observation from Table 5(b)). However almost all of the concepts are recognized by SW algorithm and nearly half of them are not recognized at all in the HHMM algorithm. It also shows that the predicted concepts in both HHMM and SW algorithm are mostly uniform (have few fragmented or merged predictions). These observation is clearly shown in our detection and uniformity property in Table 3. Our proposed metric also correctly shows the quality of detecting the boundaries of concepts while Table 5 (b) display these information totally. Since the duration of this class is much less than the total duration of this dataset while this class constitutes 13% of concepts in this dataset, the last four errors in Table 5 (b) are close to zero. Relative duration properties in Table 3 shows SW either recognize a whole ground truth concept (recall = 0.92) or does not recognize the concept at all; however, its prediction exceed the boundaries (precision < 0.6).

Table 4 shows the metric proposed in [26] with the different parameters. We can observe that γ function, which considers fragmented and merged predictions, has a small affect on the recall and precision. As it is observable from our uni-

² We use feature extraction in [13] and three layers perceptron for classifier step.

³ The internal steps are not important since the concentration is on the metrics.

⁴ For saving the space, the analysis of other classes are existed in our repository.

⁵ If the used segmentation algorithm generates more segments for longer events which is the case with the well-used sliding window method.

Table 3. Our metric and the spider chart of f1 over two algorithms for one class.

Algorithm Property	HHMM			SW		
	recall	precision	f1	recall	precision	f1
detection	0.53	0.51	0.52	0.97	0.49	0.65
uniformity	0.95	0.97	0.96	0.86	0.89	0.88
total duration	0.19	0.32	0.24	0.80	0.41	0.55
relative duration	0.39	0.58	0.47	0.92	0.54	0.68
boundary start	0.70	0.63	0.66	1.00	0.48	0.65
boundary end	0.86	0.54	0.66	0.92	0.34	0.49

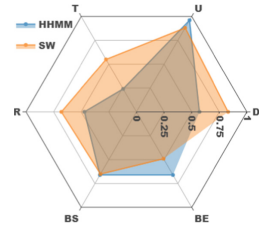


Table 4. Tatbul metric [26] with several parameters and its f1 chart for one class.

Algorithm Parameter	HHMM			SW		
	recall	precis.	f1	recall	precis.	f1
$\alpha=0, \gamma=1, \delta=back$	0.42	0.29	0.34	0.93	0.27	0.42
$\alpha=0, \gamma=1, \delta=middle$	0.39	0.37	0.38	0.92	0.36	0.52
$\alpha=0, \gamma=1, \delta=front$	0.37	0.37	0.37	0.92	0.34	0.50
$\alpha=0, \gamma=1, \delta=flat$	0.39	0.33	0.36	0.92	0.31	0.46
$\alpha=1, \gamma=1, \delta=flat$	0.53	0.33	0.41	0.97	0.31	0.47
$\alpha=0, \gamma=reci, \delta=flat$	0.39	0.33	0.36	0.92	0.30	0.45

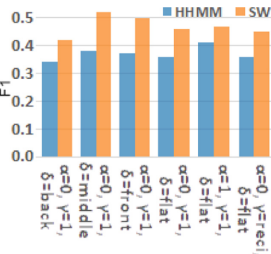


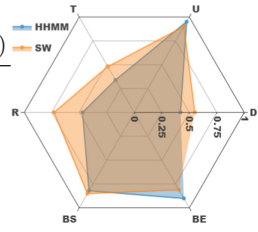
Table 5. Ward’s proposed metrics for evaluating two algorithms for one class

(a) Event metrics	HHMM	SW	(b) Time metrics	HHMM	SW
Deletions / $ R $	0.47	0.03	True positive rate	0.19	0.80
Merged / $ R $	0.03	0.13	Deletion rate	0.50	0
Fragmented / $ R $	0	0.04	Fragmenting rate	0	0.16
Frag. and merged / $ R $	0	0	Start underfill rate	0.29	0
Correct / $ R $	0.51	0.80	End underfill rate	0.02	0.03
Insertions / $ P $	0.40	0.28	1-false positive rate	1.00	1.00
Merging / $ P $	0.02	0.05	Insertion rate	0	0
Fragmenting / $ P $	0	0.06	Merge rate	0	0
Frag. and merging / $ P $	0	0	Start overfill rate	0	0
Correct / $ P $	0.58	0.61	End overfill rate	0	0

formity property in Table 3, we can see the predictions of both algorithms are uniform but HHMM works better. This observation, can not be captured from Tatbul’s metric. As analysed at the end of Sect. 2, the main issue of Tatbul’s metric is that recall and precision are not calculated in similar model and can not be used as complementary (e.g., changing α parameter has effect only on recall.). Lastly, δ parameter in Table 4 is proposed by them to consider the boundary alignment errors; however, changing that does not provide significant changes in

Table 6. Macro average of all classes by our metric over two algorithms.

Algorithm Property	HHMM (macro avg)			SW (macro avg)		
	recall	precision	f1(m)	recall	precision	f1(m)
detection	0.44	0.42	0.41	0.86	0.34	0.51
uniformity	0.98	0.92	0.95	0.97	0.85	0.9
total duration	0.31	0.46	0.34	0.58	0.4	0.48
relative duration	0.37	0.87	0.47	0.67	0.78	0.73
boundary start	0.8	0.92	0.82	0.92	0.83	0.85
boundary end	0.94	0.89	0.9	0.89	0.79	0.81



recall and precision while our boundary properties in (Table 3) clearly provide the situation of predictions. This experiment ends with Table 6 that compares the macro average of our metric across all classes of this dataset.

5 Conclusions

In general, activity events are durative in AR. Choosing an appropriate evaluating metric is an essential step to compare AR systems. However, due to the absence of an appropriate one, researchers often use time-frame, event-based, or classifier performance, which can misleadingly present convincing performance for an AR system. This paper proposes a new mathematical model to evaluate AR algorithms which is expressive (by capturing several properties of AR algorithm such as detection, total duration, relative duration, boundary alignment, and uniformity), customizable (the adjustable parameters can support a wide range of applications and can give more weights to some properties of AR algorithms), extensible (adding a new property is straightforward and independent of others). Although our method can give more meaningful information about AR algorithms, its computation complexity remains linear on the size of predictions and ground truths. Our metric has been tested on several datasets, and its ability to measure different AR algorithm properties has been shown. One exciting outcome of this formulation is the possibility to generate a profile (in terms of properties) for each algorithm. Therefore, it can be used as a heuristic for faster algorithm selection which will be explored more in future researches. We are also interested in including fuzziness in our properties.

References

1. Alemdar, H., Tunca, C., Ersoy, C.: Daily life behaviour monitoring for health assessment using machine learning: bridging the gap between domains. *Pers. Ubiquit. Comput.* **19**(2), 303–315 (2014). <https://doi.org/10.1007/s00779-014-0823-y>
2. Asghari, P., Soleimani, E., Nazerfard, E.: Online human activity recognition employing hierarchical hidden Markov models. *J. Ambient. Intell. Humaniz. Comput.* **11**(3), 1141–1152 (2020). <https://doi.org/10.1007/s12652-019-01380-5>

3. Awad, G., et al.: TRECVID 2020: a comprehensive campaign for evaluating video retrieval tasks across multiple application domains. In: Proceedings of TRECVID, pp. 1–55. NIST, USA (2021)
4. Bilen, C., Ferroni, G., Tuveri, F., Azcarreta, J., Krstulovic, S.: A framework for the robust evaluation of sound event detection. In: ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 61–65 (2020). <https://doi.org/10.1109/ICASSP40776.2020.9052995>
5. Cook, D.J., Narayanan, C.K.: Activity Learning: Discovering, Recognizing, and Predicting Human Behavior from Sensor Data. Wiley Series on Parallel and Distributed Computing, 1st edn. Wiley (2015)
6. Dutta, J., Banerjee, B.: Online detection of abnormal events using incremental coding length. In: AAAI Conference on Artificial Intelligence (2015). <https://ojs.aaai.org/index.php/AAAI/article/view/9799>
7. Fu, T.C.: A review on time series data mining. Eng. Appl. Artif. Intell. **24**(1), 164–181 (2011). <https://doi.org/10.1016/j.engappai.2010.09.007>
8. Gjoreski, H., et al.: Competitive live evaluations of activity-recognition systems. IEEE Pervasive Comput. **14**(1), 70–77 (2015). <https://doi.org/10.1109/MPRV.2015.3>
9. Hein, A., Kirste, T.: Generic performance metrics for continuous activity recognition. In: Bach, J., Edelkamp, S. (eds.) KI 2011. LNCS (LNAI), vol. 7006, pp. 139–143. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-24455-1_13
10. Hwang, W.S., Yun, J.H., Kim, J., Kim, H.C.: Time-series aware precision and recall for anomaly detection. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 2241–2244. ACM, New York (2019). <https://doi.org/10.1145/3357384.3358118>
11. Ionescu, R.T., Khan, F.S., Georgescu, M.I., Shao, L.: Object-centric auto-encoders and dummy anomalies for abnormal event detection in video. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2019-June, pp. 7834–7843. IEEE (2019). <https://doi.org/10.1109/CVPR.2019.00803>, <https://ieeexplore.ieee.org/document/8954309/>
12. Kasteren, T.V., Alemdar, H., Ersoy, C.: Effective performance metrics for evaluating activity recognition methods. In: ARCS (2011)
13. Krishnan, N.C., Cook, D.J.: Activity recognition on streaming sensor data. Pervasive Mobile Comput. **10**(PART B), 138–154 (2014). <https://doi.org/10.1016/j.pmcj.2012.07.003>
14. Lavin, A., Ahmad, S.: Evaluating real-time anomaly detection algorithms - the Numenta Anomaly Benchmark. In: IEEE 14th International Conference on Machine Learning and Applications (ICMLA), pp. 38–44. IEEE (2015). <https://doi.org/10.1109/ICMLA.2015.141>, <http://ieeexplore.ieee.org/document/7424283/>
15. Lu, Y., Kumar, K.M., Nabavi, S.S., Wang, Y.: Future frame prediction using convolutional VRNN for anomaly detection. In: IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). IEEE (2019). <https://doi.org/10.1109/AVSS.2019.8909850>
16. Mesaros, A., Heittola, T., Virtanen, T.: Metrics for polyphonic sound event detection. Appl. Sci. (Switzerland) **6**(6) (2016). <https://doi.org/10.3390/app6060162>
17. Minnen, D., Westeyn, T.L., Starner, T., Ward, J.A., Lukowicz, P.: Performance metrics and evaluation issues for continuous activity recognition. In: Performance Metrics for Intelligent Systems, pp. 141–148. NIST, Gaithersburg (2006)

18. Modaresi, S., Osmani, A., Razzazi, M., Chibani, A.: Multimodal evaluation method for sound event detection. In: IEEE International Conference on Acoustics, Speech and Signal Processing, (ICASSP). IEEE (2022)
19. Ni, Q., García Hernando, A., de la Cruz, I.: The elderly’s independent living in smart homes: a characterization of activities and sensing infrastructure survey to facilitate services development. *Sensors* **15**(5), 11312–11362 (2015). <https://doi.org/10.3390/s150511312>
20. NIST: TRECVID 2004 Evaluation (2004). <https://www-nlpir.nist.gov/projects/tv2004/index.html>
21. Osmani, A.: STCSP: a representation model for sequential patterns. *Foundations and Applications of Spatio-Temporal Reasoning (FASTR)* (2003). <https://www.aaai.org/Library/Symposia/Spring/2003/ss03-03-010.php>
22. Perera, C., Zaslavsky, A., Christen, P., Georgakopoulos, D.: Context aware computing for the Internet of Things: a survey. *IEEE Commun. Surv. Tutor.* **16**(1), 414–454 (2014). <https://doi.org/10.1109/SURV.2013.042313.00197>, <http://ieeexplore.ieee.org/document/6512846/>
23. Qian, H., Pan, S.J., Miao, C.: Latent independent excitation for generalizable sensor-based cross-person activity recognition. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 13, pp. 11921–11929 (2021). <https://ojs.aaai.org/index.php/AAAI/article/view/17416>
24. Ross, R.J., Kelleher, J.: Accuracy and timeliness in ML based activity recognition. In: *Proceedings of the 13th AAAI Conference on Plan, Activity, and Intent Recognition, AAAIWS’13-13*, vol. WS-13-13, pp. 39–46. AAAI Press (2013). <https://doi.org/10.5555/2908241.2908247>
25. Stowell, D., Giannoulis, D., Benetos, E., Lagrange, M., Plumbley, M.D.: Detection and classification of acoustic scenes and events. *IEEE Trans. Multimedia* **17**(10), 1733–1746 (2015). <https://doi.org/10.1109/TMM.2015.2428998>
26. Tatbul, N., Lee, T.J., Zdonik, S., Alam, M., Gottschlich, J.: Precision and recall for time series. In: *Neural Information Processing Systems (NIPS)* (2018). <https://papers.nips.cc/paper/7462-precision-and-recall-for-time-series>
27. Ward, J.A., Lukowicz, P., Gellersen, H.W.: Ward: performance metrics for activity recognition. *ACM Trans. Intell. Syst. Technol.* (2011). <https://doi.org/10.1145/1889681.1889687>