Towards New Performance Metrics for Multi-level Data for Activity Recognition

Sayeda Shamma Alia, Paula Lago and Sozo Inoue

Abstract Although activity recognition datasets are usually labeled with only one activity per timestamp, semantic annotations could be given with multiple granularity levels. For example, we can describe both the current activity and the current step within that activity (i.e. cooking and taking ingredients from fridge). However, it is challenging to evaluate the performance of the classifiers under this condition. This is because the evaluation has to consider many underlying challenges such as the dependence of one level to the other, the direction of this dependence, the impact of each level, data imbalance for each level and in between levels. These factors have a great impact for assessing performance of classifiers. This work proposes two metrics for multi-level labels in a dataset considering these factors. We compare the metrics on a public dataset and show that they can assess the performance of a model classifying activities at two different granularity levels. Among the proposed metrics, a metric considering the co-occurrence matrix of correct predictions for both levels, shows the best discrimination policy, with a difference between two classifier models of about 12%.

1 Introduction

Activity recognition is the task of classifying daily life activities performed by a human from the readings of one or multiple sensors[1]. Although it may sound easy, there are many underlying challenges due to the complex nature of activities in real life settings. One such challenge is that many activities are composed of a number of

Sayeda Shamma Alia
Kyushu Institute of Technology, e-mail: alia@sozolab.jp

Paula Lago
Kyushu Institute of Technology, e-mail: paula@mns.kyutech.ac.jp

Sozo Inoue
Kyushu Institute of Technology, e-mail: sozo@brain.kyutech.ac.jp

small activities. For example, COOKING is composed of numerous small tasks such as TAKE from a drawer or pantry, or MIX ingredients in a bowl. These small tasks will have wide variation depending on the specific recipe that is being prepared, the time of day, or the person who is doing them. In this work, complex activities will be named as *macro activity* and the small activities that compose them will be called *micro activity*.

Annotating datasets for complex activities can create multi-level labels, where one instance is labeled with both its macro and micro activities. When considering multi-level labels, it is not only difficult to classify the activities correctly, it is also very difficult to measure the performance of the classification algorithm. Existing metrics such as F-score, accuracy measure performance considering that the dataset has a single label for each instance. However, it is possible to annotate complex activities with multi-level labels to represent multiple semantic levels (activities and actions). To accurately measure the quality of complex activity recognition, a new performance metric is needed which can assess the classifier quality for both levels.

In this work, we discuss the challenges for designing a performance metric for multi-level labels. After that, we propose and compare two different metrics for this purpose, each suited for different scenarios and goals. All of the proposed metrics showed good result, but 2nd proposed metric seems to distinguish better than other two having difference of the algorithms in performance about 12%.

2 Related Work

Although multiple studies related to complex activity recognition are found in the literature[10], [9], some works [11], [12], [13] also consider that the annotation of complex activities can be given at multiple granularity levels. But, they use different performance metric for different levels for evaluation. Traditional classification performance metrics such as accuracy, F-measure, or AUC are used in most works. These performance metrics can measure performance for one level of annotations, but they are unable to measure performance for data annotated with multi-level labels. So, performance evaluation is done on different level individually.

Activity recognition is becoming important day by day for its various application in different fields [2]. Li et. al.[10] assert that complex activities consist of atomic activities and their temporal dependencies. For example, relax can have: walk, reach lazy-chair, sit, lie down, and drink from cup. These activities have some relation with each other. They also suggest some uncertainty associated with the atomic activities and the sequences of these activities can vary from person to person. In this work they analyzed the atomic activities and their temporal relation for complex activity recognition. Based on Allen interval relations they proposed a generative probabilistic model so that it can handle the inter variability and relation of the complex data. Then they predicted the complex activities and compared their results with four algorithms and showed that their method outperformed others. For

measuring the performance of the classifiers, they used accuracy for whole dataset and error rate for atomic activities.

In [9], authors identified both simple and complex activities. As a means for data collection they emphasized the use of smartphones presenting more natural scenario rather than other sensors in controlled environment. Simple activities were: biking, climbing stairs, driving, lying, running, sitting, standing and walking. Complex activities were: cleaning, cooking, Medication, Sweeping, Washing Hands, Watering Plants. The classifiers used for this work are: Multi-layer Perceptron, Naïve Bayes, Bayesian network, Decision Table, Best-First Tree and K-star. For complex activities, Multi-layer Perceptron performed best and it was about 50%. In this work, they did not consider the micro level of the complex activities.

The authors in [4] proposed a model named InnoHAR for complex activity recognition. Inception Neural Network and recurrent neural network are combined for the proposed work. They applied the method on 3 datasets and used F-measure as their performance metric. Comparing with four methods, proposed method performed best for all the datasets.

In [3], authors proposed a novel Context-Driven Activity Theory (CDAT) based on probabilistic and Markov chain analysis for complex activity recognition. In this case, atomic activities are considered and used to create complex activity signatures. Based on these signatures, definition for each complex activity is obtained. For constructing knowledge base, domain knowledge and experiment data from real-life scenario is used. There are 16 complex activity classes and the performance was compared with decision tree and J48 algorithm. Accuracy was used as performance measure for this work. They showed that their method can significantly reduce the percentage of data needed for training.

Authors in [14] recognized high level activities based on multiple mid-level activities and atomic actions. They named the later two as activity concepts. These activity concepts are used to bridge the gap between low level features and high level activities. They used video dataset and used Fisher kernel techniques to encode activity concept transitions and then used Hidden markov model. The idea behind this, is to prevent loss of temporal information from low level features. Although they mentioned high level and low level activities, but for classified high level activities using low level actions. For performance measurement average precision was used.

In [13], Hierarchical hidden markov models are proposed to recognize multilevel activities. They considered two levels and referred lower level activities as "Action" and higher level activities as "Activity". They collected dataset for their work and there were three activities and four actions. Precision was used for assessment for activity and action recognition individually.

The authors in [12] considered four different levels of activity, namely: complex activities, simple activities, manipulative gestures, atomic gestures. They put an emphasis on recognition of both coarse-grained and fine-grained level of detail. They proposed probabilistic description logics (DLs) to detect multilevel activities. To represent the multilevel activities they used ontological reasoning. In [11] they reported the results of the method they proposed in [12]. Although their method were

able to perform very good, they evaluated performance individually for different levels. Precision, recall and F-score was used for measurement.

The mentioned works considered complex activity with atomic activities for classification and also considered the relationship. Some of the works used low level actions to detect high level activities[14], [12], [16]. To evaluate the performance of classifier, use of different performance metrics are observed, like: average precision [14],[15], intersection-over-union (IoU) [15], precision, recall and f-score. But, different performance metric for different levels are used for evaluation. Dependency of the levels, direction of the dependency, imbalance in different levels are not considered. So, a new performance metric is needed which can take account of these factors and judge a classifier for multi-level labeled dataset properly.

3 New Performance Metrics for Multi-level Labels

In this section, we discuss two approaches to define a new performance metric to consider multi-level labels during evaluation. Also, one baseline performance metric is discussed in first part of this section. To illustrate how each approach works, we will consider a scenario with 6 test instances (Table 1). In this table, the values of zero and one in the macro column represent an incorrect or correct classification respectively. In the micro column, the first digit represents the number of correctly predicted labels (true positives + true negatives) and the second digit means the actual number of labels for that instance. For example, for instance number 1, the macro activity was correctly predicted and 3 out of 3 micro activities were correctly predicted, whereas for instance number 5, the macro activity was not correctly predicted and 2 out of 3 micro activities were correctly predicted. Using this example table all the approaches will be described.

Table 1: Use case scenario to illustrate the proposed metrics. The table shows the number of correct predictions for each label level.

Instance Id.	Macro	Micro
1	1	3/3
2	1	2/3
3	1	0/2
4	0	3/3
5	0	2/3
6	0	0/2

For the calculations and definitions in this section, we use the following definitions:

- N the total number of instances. $N = 6$ in the example.

- TP_{m_a}, the number of true positive instances in the macro level. $TP_{m_a} = 3$ in the example.
- $TP^j_{m_a}$, the number of true positives for instance j in the macro level. If the macro level has only one possible label for each instance, as we assume in this paper, this number can only be 0 or 1.
- $TP^j_{m_i}$ the number of correctly predicted micro activity labels for instance j. In the example $TP^1_{m_i} = 3$ and $TP^5_{m_i} = 2$.
- M^j_i the number of micro labels assigned to instance number j. In the example, $M^1_i = 3$ and $M^3_i = 2$

We now describe the one baseline and two proposed metrics. To propose the different metrics, we consider the following:

- Whether the macro and micro activities are independent from each other. The levels are considered independent of each other if knowing one of the labels does not provide further information about the other levels. In other words: $P(m_i) = P(m_i|m_a)$ where m_i and m_a are the micro and macro activities. This independence can be assumed to simplify the measurement.

3.1 Baseline Metric: Average

When independence between the two label levels is assumed, then each level performance can be evaluated independently ($P(m_a)$ and $P(m_i)$) and an `OR` relation can be used for assessing the combined performance of the classifiers (Eq. 1). In other words, a simple average of the performance on each level is taken as the combined evaluation metric.

$$P_{avg} = \frac{P(m_a) + P(m_i)}{2} \times 100\% \tag{1}$$

Note that $P(m_a)$ and $P(m_i)$ can be measured with any conventional metric (Accuracy,F-score,G-means) and therefore, P_{avg} can have multiple meanings depending on the chosen metric.

For the example illustrated in Table 1, we calculate the performance of macro and micro levels separately with the accuracy metric. In this scenario, P_{avg} can be calculated as in Eq. 2:

$$P_{avg} = \frac{P(m_a) + P(m_i)}{2} \times 100\% \tag{2}$$

where

$$m_a = \frac{TP_{m_a}}{N} \tag{3}$$

$$m_i = \frac{1}{N} \sum_{j=1}^{j=N} \frac{TP^j_{m_i}}{M^j_i} \tag{4}$$

For the example in Table 1, calculation will be like below:

$$\begin{aligned} P_{avg} &= \frac{m_a + m_i}{2} \times 100\% \\ &= \frac{\frac{3}{6} + \frac{\frac{10}{3}}{6}}{2} \times 100\% \\ &= 52.8\% \end{aligned}$$

3.1.1 Impact of each level

This metric considers that both levels have the same importance. However, if the importance of the levels are different, then weights can be assigned for each level. The weight can be defined as α, so as to make a weighted average as in Eq. 5. The value of α can be set manually.

$$P_{w\_avg} = (m_a \times \alpha + m_i \times (1 - \alpha)) \times 100\% \tag{5}$$

If the value of α is 0.5, then it is the same as Eq.2.

3.2 Proposed metric 1: Considering dependency of the levels

For multi-level labeled data, the levels can depend on each other. We consider three possible directions of the dependence relation:

- Macro level to Micro level: This relation indicates that incorrect predictions at the micro-level will yield the whole prediction incorrect. For example, in a health care scenario, if a nurse does not sanitize their hands before touching patient for blood test, it can spread germs. One very evident example is outbreak of Coronavirus[17]. Therefore, if we cannot identify the step hand-wash during the activity blood-test, the prediction is irrelevant because we cannot evaluate the correct completion of the activity.
- Micro level to Macro level: This relation indicates that an incorrect prediction at the macro-level makes the combined prediction as incorrect. In a cooking scenario, this relation holds because the way of doing activities can vary from person to person. Some micro activities can be skipped, for example during 'making tea', adding milk and sugar is optional. Detecting the macro activity is more important in this scenario, because micro activities can change.
- Two-way relationship: This relation indicates that an incorrect prediction at either level makes the combined prediction as incorrect.

In the following metrics, we use the definition of *instance accuracy* (acc^j) as the average of the micro and macro level accuracy of each level for the instance (Eq. 6).

$$acc^j = \frac{1}{2} \times \left(TP_{ma} + \frac{TP^j_{mi}}{M^j_i} \right) \tag{6}$$

The macro-level depends on the micro-level When the macro level depends on the micro level, we penalize incorrect predictions at the micro level. For this, whenever an instance has no correct predictions at the micro activity level, its accuracy is set to zero. We use the indicator function $\mathbb{1}_{(TP^j_{mi}>0)}$ for this.

In this scenario, the overall performance P_{mi} is calculated as in Eq. 7.

$$P_{mi} = \frac{1}{N} \sum_{j=1}^{N} \mathbb{1}_{(TP^j_{mi}>0)} * acc^j \tag{7}$$

In the example from Table 1, sample 3 and 6 will be omitted and performance will be calculated as follows:

$$\begin{aligned} P &= \frac{1 + \frac{3}{3} + 1 + \frac{2}{3} + 0 + \frac{3}{3} + 0 + \frac{2}{3}}{6 \times 2} \times 100\% \\ &= 44.4\% \end{aligned}$$

The micro-level depends on the macro-level: This case is the opposite of the above assumption, therefore we set the instance accuracy when there are no correct predictions at the macro activity level. The overall performance P_{ma} is calculated as in Eq. 8.

$$P_{ma} = \frac{1}{N} \sum_{j=1}^{N} TP^j_{ma} * acc^j \tag{8}$$

In this scenario only the first 3 instances will be taken for calculation. The calculation will be like below:

$$\begin{aligned} P &= \frac{1 + \frac{3}{3} + 1 + \frac{2}{3} + 1 + \frac{0}{2}}{6 \times 2} \times 100\% \\ &= 38.9\% \end{aligned}$$

Both levels depend on each other: When an incorrect prediction at both levels is equally important, then we set the instance accuracy to zero when either level is not correctly classified. The overall performance is then calculated as in Equation 9.

$$P_{ma\_mi} = \frac{1}{N} \sum_{j=1}^{N} TP^j_{ma} * \mathbb{1}_{(TP^j_{mi}>0)} * acc^j \tag{9}$$

If both of the levels are dependent on each other, then only first 2 instances will be taken for calculation. The calculation will be like below:

$$P = \frac{1 + \frac{3}{3} + 1 + \frac{2}{3}}{6 \times 2} \times 100\%$$
$$= 30.6\%$$

3.3 Proposed metric 2: Co-occurrence matrix

A co-occurrence matrix is usually used to assess classification performance, as it can provide more information than a single metric. It can also be a good indicator to assess performance for multi-level dataset as it can provide a more complete picture of the classifier performance for both levels. A traditional co-occurrence matrix, however, cannot be used for multi-level labels because it considers only one level as other traditional metrics. We define a co-occurrence matrix for multi-level labels by setting the rows to represent the number of correct predictions in the macro level and the columns to represent the number of correct predictions in the micro level.

As shown in Figure 1, row name 0 and 1 means incorrect and correct prediction respectively. For micro level, total number of labels can be divided by 2 and will be in Mi1 and Mi2. For example, if there are 5 labels, then Mi1 will be (0-3) and Mi2 will be (4-6). The cells in the table will be occurrences of that labels. In this figure the meaning of each cell is given below:

- **A=** Occurrences when macro is incorrectly predicted and 0 to 3 labels are predicted correctly in micro.
- **B=** Occurrences when macro is incorrectly predicted and 4 to 6 labels are predicted correctly in micro.
- **C=** Occurrences when macro is correctly predicted and 0 to 3 labels are predicted correctly in micro.
- **D=** Occurrences when macro is correctly predicted and 4 to 6 labels are predicted correctly in micro.

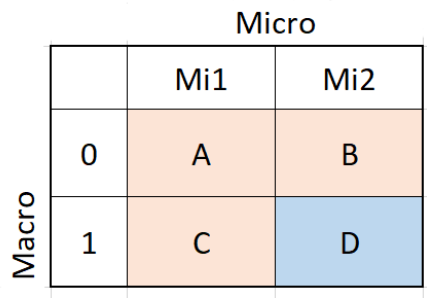

Fig. 1: Co-occurrence Matrix for Proposed performance Metric

Performance of a classifier can be calculated like below:

$$P = \frac{D}{A + B + C + D} \times 100\% \tag{10}$$

Here, D is the closest to actual label and most important as it represents correct prediction in macro and maximum correct prediction in micro. The sum of A, B, C and D is N. The ratio is taken because it reflects how much a classifier is closer to actual prediction.

4 Evaluation

In this section we use all three metrics and other traditional metrics to evaluate two classification models for a multi-level labeled dataset. We first describe the dataset used for the experiment, features used and classifier models. Then we show the results of the comparison.

4.1 Dataset Description

A sensor-based activity recognition dataset with micro and macro activities in a cooking scenario [5] is used to compare the metrics proposed in this work. The dataset consists of cooking activities with recipes as macro activities (3 recipes) and steps as micro activities (9 possible steps). Data is segmented into 30-second segments, and labels are given for each segment. Most micro activities take less than 30 seconds, so there are multiple labels for micro activities in most of the segments. The data was collected in a controlled environment where subjects had to prepare three types of foods using pre-defined recipes. Data was collected using 2 smart phones placed in right arm and left hip, 2 smart watches placed in both of the wrist of a subject and motion capture system using 29 markers. In this experiment only smart watch and smart phone data are used. The data was collected using 4 subjects. Among them 3 subjects were used for training and 1 subject for testing. Number of samples for the classes can be seen in Table 2. It is clear that there is an imbalance in the data distribution of the classes in both the macro and micro levels. From macro activities, Cereal has comparatively less number of samples. For micro activities, Add, Mix, Open have less samples. The micro activities that compose each macro activity are:

- **Cereal:** Take , Open , Cut , Peel , Other, Put
- **Fruitsalad:** Take , Add , Mix , Cut , Peel , Other, Put
- **Sandwich:** Take , Cut , Other, Wash , Put

Table 2: Number of samples for training and testing

	Classes	Training	Testing
Macro	Cereal	219	78
	Fruitsalad	306	114
	Sandwich	339	105
Micro	Take	402	138
	Add	54	18
	Mix	57	12
	Open	69	18
	Cut	297	93
	Peel	288	108
	Other	222	102
	Wash	90	30
	Put	342	138

4.2 Feature Extraction

The cooking dataset is originally divided into 30 second time windows. For feature extraction, each time window was divided into 3 segments in this work, making it into 10-second time windows. Then features were extracted from these windows. The main goal of this work is to propose the performance metric, therefore basic features were used. The features used are: mean, standard deviation, variance, max, min, median, kurtosis and acceleration difference. Here 6 acceleration difference were calculated by taking the difference between the mean acceleration in each window for the pairs indicated:

- dRWristArm: Acceleration difference of Right Wrist to Right Arm
- dRWristLWrist: Acceleration difference of Right Wrist to Left Wrist
- dRWristLHip: Acceleration difference of Right Wrist to Left Hip
- dRArmLWrist: Acceleration difference of Right Arm to Left Wrist
- dRArmLHip: Acceleration difference of Right Arm to Left Hip
- dLWristHip: Acceleration difference of Left Wrist to Left Hip

We the use Info gain[7](Figure 2) and Chi-square[6] (Figure 3) metrics to analyze the importance of each feature. We observe in the figures that the acceleration difference has a great impact on classification.

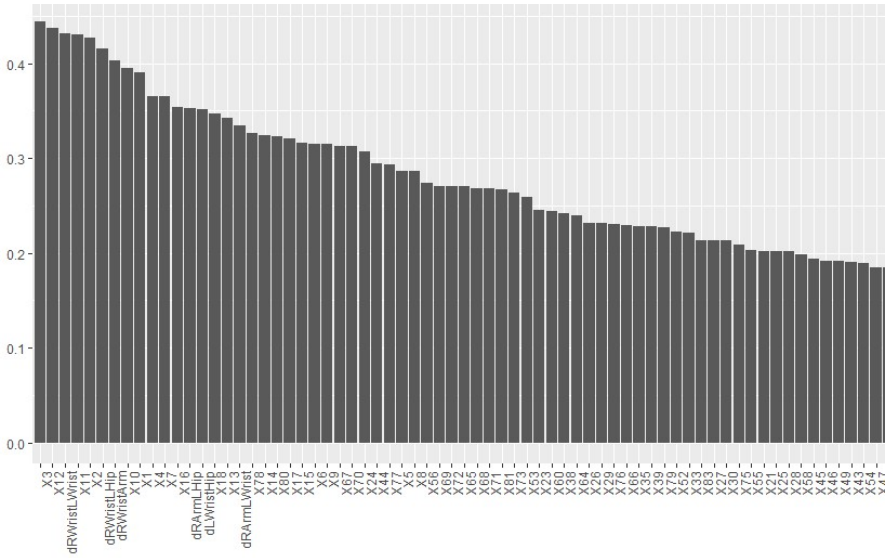

Fig. 3: Importance of Acceleration Difference feature using Chi-squared

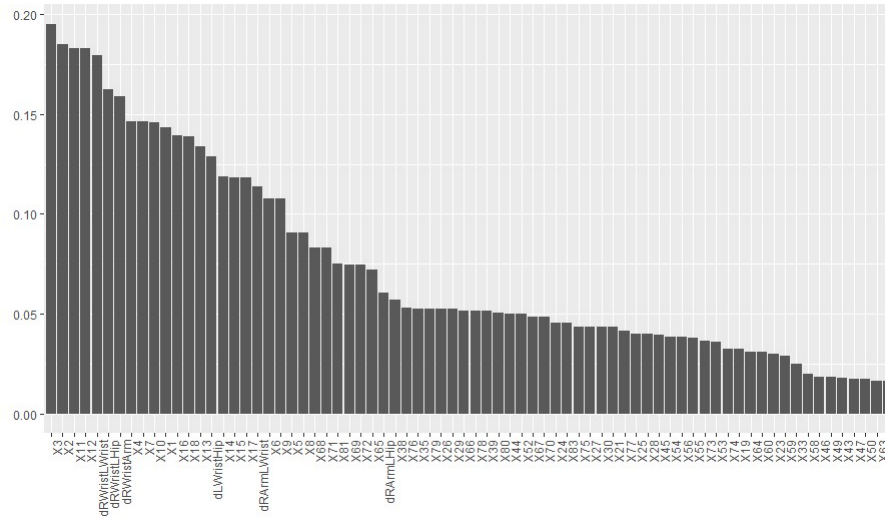

Fig. 2: Importance of Acceleration Difference feature using Information Gain

4.3 Classification Methods

As it is not the scope of this work to propose or develop a new classifier for multi-level scenario, we use different classifier for different levels for classification. Details of the used classifiers are given in following sections.

4.3.1 Macro Activity Classification

Traditional classification algorithms: Random Forest, Decision Tree, Naive Bayes, kNN, Support Vector Machine were used for classification of Macro Activities. The results are shown in Table 3 for this level. From the table it seems that precision is comparatively higher than recall in most of the cases. Random forest has overall a good result both in Precision and Recall. For this dataset in Cooking scenario, high precision is desirable. But in cases like: fall detection, high recall or less number of true negatives are desired. Some worst performance of some classifier can be seen in CEREAL class. The reason behind this is further discussed in Section 5.2.

Table 3: Classifiers' performance for Macro level

Method Name	Performance Metric	Class Names		
		CEREAL	FRUITSALAD	SANDWICH
KNN	Precision	37.4	48.0	40.4
	Recall	**43.6**	43.0	40.0
Random Forest	Precision	48.5	48.1	**47.2**
	Recall	41.0	43.9	57.1
Decision Tree	Precision	32.1	45.9	40.0
	Recall	23.1	**58.8**	36.2
Naive Bayes	Precision	34.2	**48.5**	40.5
	Recall	34.6	57.0	32.4
SVM	Precision	**50.0**	46.7	46.0
	Recall	43.6	36.8	**61.0**

4.3.2 Micro Activity Classification

As discussed before, each segment has multiple micro activity labels due to the duration of the segments. For this reason, multi-label algorithms were used from the mlr package [8] in R, namely: Random Forest and Random Ferns. Performance of these two classifiers are shown in Table 4. In the table, ADD , MIX , and OPEN have precision and recall of 0% for Random Forest. One reason for this result might be the low number of training samples for these classes as seen in Table 2. Overall, Random Ferns seems to perform better than Random Forest. The reason behind the good performance will further be analyzed in Section 4.4.

Table 4: Classifiers' performance for Micro level

Method Name	Metric	Class Names								
		Take	Add	Mix	Open	Cut	Peel	Other	Wash	Put
Random Forest	Precision	88.2	0.0	0.0	0.0	62.1	77.1	48.4	0.0	73.4
	Recall	81.2	0.0	0.0	0.0	58.1	50.0	14.7	0.0	50.0
Random Ferns	Precision	80.8	9.1	6.1	13.0	51.3	64.7	50.0	12.9	73.0
	Recall	85.5	72.2	75.0	77.8	84.9	71.3	42.2	66.7	64.5

4.4 Performance Evaluation with Proposed Metrics

The previous section analyzed the performance for macro and micro activity classification using traditional metrics. As could be seen, these metrics assess the performance for only one level, but we cannot evaluate the performance of the combined algorithms.

We discussed several approaches and factors in Section 3 for assessing the quality of the algorithms for 2-level datasets. To calculate these metrics for the classifiers evaluated, we first calculate the co-occurrence matrix, as described in Section 3.3. Table 5 shows this matrix. In this table, each row shows the result of the macro-level: 0 and 1 means incorrect and correct prediction respectively. The columns represent the number of correctly predicted labels for micro activities. Notice that this means both true positives and true negatives. Finally, each cell represents the frequency and the sum column summarizes the number of correct and incorrect instances for the macro-activity level. As an example, in column 5, the first cell reads '10' which means that there were 10 instances in the test data where the macro label was not correctly predicted and 5 micro labels were correctly predicted.

Table 5: Occurrence matrix of macro and micro activity classes

		0	1	2	3	4	5	6	7	8	9	Sum
Random Forest	0	0	0	0	1	4	10	22	36	55	27	155
	1	0	0	0	0	1	6	16	32	47	40	142
Random Ferns	0	0	2	1	12	29	35	37	28	20	7	171
	1	0	1	4	1	15	21	28	18	25	13	126

The results of the performance evaluation with the proposed metrics (Section 3) are shown in Table 6. Classifier 1 refers to a method using Random Forest for both macro and micro activity classification and Classifier 2 refers to a method using Naïve Bayes for macro and Random Ferns [8] for micro activity classification. Other combinations of the algorithms evaluated in the previous section could be used but we chose these two to show the behavior of the different metrics.

From the results, we can see that in all cases, classifier 1 outperformed classifier 2. Using proposed metric 2, however, we observe a larger difference between the performance of classifier 1 and classifier 2. From Table 5 we can understand that classifier 1 has a larger number of correct predictions both for micro and macro level. Therefore, we understand that proposed metric 2, has a better discrimination for the best classifier.

Table 6: Performance of classifiers using different performance metric

	Classifier 1	Classifier 2
Baseline Metric	65.54%	54.13%
Proposed metric 1	44.29%	35.95%
Proposed metric 2	47.47%	35.35%

5 Discussion

This work proposed two performance metrics for evaluation of multi-level labeled dataset considering dependency of levels and using co-occurrence matrix. We have compared the performance of two classifiers on the Cooking activity dataset which is labeled with two levels of activities. The proposed metrics are more suited for multi-level labeled dataset because of following:

- Consider dependence between the levels
- Consider the direction of dependencies
- Put emphasis on prediction that is more close to actual label

From the results, we observe that the proposed metrics agree in which is the best algorithm but the differences become larger with the second metric. Also the second metric considers the relation of the levels better and understands the classifiers better. For example, in a scenario where 100% of the macro levels are correctly predicted and none of the micro levels are correctly predicted or vice-versa using an algorithm, the baseline metric will evaluate that the algorithm is able to predict 50% of the levels correctly. But actually it completely failed to recognize micro levels. So, this insight is important to observed when designing performance metric for multi-level labels. The proposed metrics are able to give these insights of the algorithms. With proposed metric 1, based on the dependency of the levels and direction of the direction of dependencies, different result will be observed for the given scenario. If the detection of macro level is more important, like detecting mode of transportation. In this case, even if an algorithm fails to detect the micro labels properly, the algorithm should get higher value for performance as macro label detection is more important here.
In a situation, where patients are remotely monitored, it is very important to detect micro activities perfectly to ensure safety and good condition of patient. In this scenario, micro activity detection is more necessary than macro activity detection.

The proposed metric 1 is able to highlight these scenario, and based on the goal of the work, any variant of the performance metric 1 can be chosen.
In a case, where detection of both levels are important, proposed metric 2 should be used. Suppose, for rehabilitation patients, observing them and checking their improvement, detection of both of the macro and micro activities are very important. So, proposed metric 2 is able to detect the relation of the level more correctly. Because an algorithm will never have good performance if it fails to recognize any of the level poorly.

We will now analyze some additional factors that should also be considered in the design of performance metrics for multi-level datasets.

5.1 Impact of True Negatives and False Positives

The impact of True Positives (TP), False positives (FP) and True Negatives(TN) is not equal for every application. For example, in a cooking scenario, having a high number of FP or TN will not have a large impact if we want to recommend recipes. However, if we want to provide assistance to elderly people or to detect if person is facing some memory related disease by observing their daily activities, good recall will be crucial. Therefore, considering the effect of precision and recall, more specifically of TP, FP and TN, is important. All of the proposed metrics in this work considers these values but they give different weight to each of them. For example, Proposed performance metric 2 has bigger emphasis on TP. TP rate is called sensitivity. Sensitivity and specificity are important measure to get an insight of the performance of the algorithms. In binary classification problem, having positive and negative classes, sensitivity is calculated like following:

$$Sensitivity = \frac{TP}{TP + FP} \tag{11}$$

Specificity means true negative rate and calculated like below.

$$Specificity = \frac{TN}{TN + FP} \tag{12}$$

But calculating these values are challenging for multi-level dataset. Because, macro and micro levels can have different samples for TP, TN, FP and FN. For example, in Table 1, instance 1 is correctly predicted in both of the levels, so it falls under TP. But for instance 4, macro level is not correctly predicted, whereas the micro level is correctly predicted. This makes the calculation of TP, TN, FP and FN complex for multi-level labels. As, the proposed metrics are not evaluating the performance of the classifiers on class level, we wish do do this and solve the problem of defining sensitivity and specificity in future by doing extensive analysis.

5.2 Influence of Class Imbalance

In real scenarios, activities are bound to have different number of samples due to differences in their duration. For example, a regular person's sleeping time will definitely be longer than their eating time. When using fixed-size windows, as regularly done in activity recognition, sample size for the activity classes is bound to differ which leads to class imbalances [1]. The dataset we used for this work is an example, with imbalance in sample size for both macro and micro levels. Considering the impact of the number of samples in each of the class is necessary. Imbalance in Macro activity affects the result in the dataset. For example, Cereal and Fruitsalad have almost same micro activities but different number of samples, except for SVM all other classifier identified Fruitsalad better than Cereal (Table 3). Also imbalance in Micro class affects performance of Macro class. For example, Fruit salad and Sandwich have different number of micro activities. The micro activities in Sandwich have less imbalance compared with Fruitsalad. For this reason, SVM and Random forest have high recall than other classifiers. For now the proposed metrics are affected by class imbalance as all of the metrics considers how many of the samples are correctly predicted without class information. For example, if we consider performance metric 2 and suppose, we have 100 samples, 90 belonging to class 1 and 10 belonging to class 2 (macro labels). If an algorithm correctly predicts all of the samples of class 1 for both macro and micro labels (category "D" from Figure 1) and completely fails to recognize class 2 ((category "A" from Figure 1)), then the performance of the classifier would be

$$P = \frac{90}{100} \times 100\% = 90\% \tag{13}$$

So, the classifier will have a great performance according to performance metric 2, although it fails to predict class 2. So, imbalance needs to be addressed. We are planning to incorporate class information with the proposed metrics for future work.

6 Conclusion

In this paper we have proposed two metrics to evaluate classification methods for multi-level labels. A multi-level label occurs when the semantic meaning of the ground truth annotation can be given at different granularity levels, for example, activities and actions as are the case examined in this paper. Other examples occur in image classification, where we can annotate and classify both the object and its attributes such as color.

Measuring the performance of classification models under such condition is challenging because of the complex nature of data and other dependency factors. The metrics proposed consider dependence relations between both levels (assuming only two levels of annotation) and evaluate according to the co-occurrence of true

positives and true negatives. It is important to consider all these mentioned issues to properly judge the quality of classifiers.

In the future, we would like to study new classification models that can classify both levels at the same time, and propose metrics that consider other challenges such as imbalance and the effect of false positives.

References

[1] A Bulling, U Blanke, B Schiele (2014) A tutorial on human activity recognition using body-worn inertial sensors. ACM Computing Surveys, doi: https://doi.org/10.1145/2499621

[2] N Ravi, N Dandekar, P Mysore, M L Littman (2005) Activity Recognition from Accelerometer Data. In: American Association for Artificial Intelligence

[3] S Saguna, A Zaslavsky, D Chakraborty (2013) Complex activity recognition using context-driven activity theory and activity signatures. ACM Transactions on Computer-Human Interaction, doi: https://doi.org/10.1145/2490832

[4] C Xu, D Chai, J He, X Zhang, S Duan (2019) InnoHAR: A Deep Neural Network for Complex Human Activity Recognition. IEEE Access, doi: 10.1109/ACCESS.2018.2890675

[5] P Lago, S Takeda, K Adachi, S S Alia, M Matsuki, B Benai, S Inoue, F Charpillet (2020) Cooking Activity Dataset with Macro and Micro Activities. IEEE Dataport, doi: 10.21227/hyzg-9m49

[6] W J Dixon, F J Massey (1969) Introduction to statistical analysis. McGraw-Hill

[7] Y Yang, J O Pedersen (1997) A comparative study on feature selection in text categorization. ICML 97:412–420

[8] P Probst, Q Au, G Casalicchio, C Stachl, B Bischl (2017) Multilabel Classification with R Package mlr. The R Journal 9/1: 352–369

[9] S Dernbach, B Das, N C Krishnan (2012) Simple and Complex Activity Recognition Through Smart Phones. In: International Conference on Intelligent Environments, 18th edn. IEEE, Mexico

[10] L Liu, L Cheng, Y Liu, Y Jia, D S Rosenblum (2016) Recognizing Complex Activities by a Probabilistic Interval-Based Model . In: AAAI Conference on Artificial Intelligence, 30th edn. AAAI, USA

[11] R Helaoui, D Riboni, H Stuckenschmidt (2013) A Probabilistic Ontological Framework for the Recognition of Multilevel Human Activities. In: ACM international joint conference on Pervasive and ubiquitous computings, ACM, Switzerland

[12] R Helaoui, D Riboni, M Niepert, C Bettini, H Stuckenschmidt (2012) Towards activity recognition using probabilistic description logics. In: Activity Context Representation: Techniques and Languages ,vol. WS-12-05 of AAAI Technical Report, AAAI

[13] Y S Lee and S B Cho (2011) Activity Recognition Using Hierarchical Hidden Markov Models on a Smartphone with 3D Accelerometer. In: International Conference on Hybrid Artificial Intelligence Systems, Springer: 460–467

[14] C Sun and R Nevatia (2013) ACTIVE: Activity Concept Transitions in Video Event Classification. In: The IEEE International Conference on Computer Vision (ICCV), IEEE

[15] C Gu, C Sun, D A Ross, C Vondrick, C Pantofaru, Y Li, S Vijayanarasimhan, G Toderici, S Ricco, R Sukthankar, C Schmid, J Malik (2018) AVA: A Video Dataset of Spatio-temporally Localized Atomic Visual Actions. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE: 6047–6056

[16] S H Jung, Y Guo, H Sawhney, R Kumar (2008) Action Video Retrieval based on Atomic Action Vocabulary. In: ACM international conference on Multimedia information retrieval (MIR), ACM: 245—252

[17] Coronavirus, World Health Organization. Available via https://www.who.int/health-topics/coronavirus.