Zero-Shot Human Activity Recognition via Nonlinear Compatibility Based Method

Wei Wang
LILY, Interdisciplinary Graduate School, Nanyang Technological University
School of Computer Science and Engineering, Nanyang Technological University
Singapore
wwang008@e.ntu.edu.sg

Chunyan Miao
School of Computer Science and Engineering, Nanyang Technological University
Singapore
ascymiao@ntu.edu.sg

Shuji Hao
Institute of High Performance Computing, A*STAR
Singapore
haosj@ihpc.a-star.edu.sg

ABSTRACT

Human activity recognition aims to recognize human activities from sensor readings. Most of existing methods in this area can only recognize activities contained in training dataset. However, in practical applications, previously unseen activities are often encountered. In this paper, we propose a new zero-shot learning method to solve the problem of recognizing previously unseen activities. The proposed method learns a nonlinear compatibility function between feature space instances and semantic space prototypes. With this function, testing instances are classified to unseen activities with highest compatibility scores. To evaluate the effectiveness of the proposed method, we conduct extensive experiments on three public datasets. Experimental results show that our proposed method consistently outperforms state-of-the-art methods in human activity recognition problems.

ACM Reference format:
Wei Wang, Chunyan Miao, and Shuji Hao. 2017. Zero-Shot Human Activity Recognition via Nonlinear Compatibility Based Method. In *Proceedings of WI '17, Leipzig, Germany, August 23-26, 2017,* 9 pages.
DOI: 10.1145/3106426.3106526

1 INTRODUCTION

Recognizing human activities from sensor readings is an important component of many context-aware applications, such as elderly caring [28], fall detection [6], context-based personal assistants [22] and smart homes [30]. In past years, extensive researches have been conducted in this area [5, 8, 15] and a lot of recognizing methods have been proposed. Inspiring performances have been achieved by these methods, but there still exist some problems. One prominent problem is the ability of recognizing previously unseen activities. Most of existing methods can only recognize activities contained in training dataset, don't have the ability to recognize previously unseen activities. However, in practical applications, we can often encounter activities not contained in training data. This is because, on one hand, the number of activities an individual can perform is large; on the other hand, activities an individual performs change with time and occasions. In model training phase, it is hard to collect sensor readings for all activities we can encounter in practical applications. How to recognize previously unseen activities is a problem of practical value, but cannot be solved by most of existing methods.

The problem of recognizing previously unseen activities is generally referred to as zero-shot learning problem. In the setting of zero-shot learning, activity classes (seen classes) contained in training dataset are different from classes (unseen classes) we need to recognize in testing phase. By introducing an extra information source (semantic space), information contained in seen classes is transferred to unseen classes, and recognition of unseen activities is achieved. The introduced semantic space contains class prototypes of each class in both seen and unseen classes. In recent years, several zero-shot learning methods [9, 10, 35] have been proposed in activity recognition area. However, these methods have some limitations. In [9] and [10], the optimization in training phase is for each dimension of the semantic space independently, not for the whole classification process. In [35], the unseen classes to recognize need to be known in training phase. The model is trained for a fixed set of unseen classes. If there are new unseen classes to recognize, they have to retrain a new model. Also the semantic space they used in their method is got from text descriptions in Web pages. But in practice, there are lots of activities which we cannot easily find appropriate Web text descriptions.

In this paper, we propose a novel zero-shot learning method for human activity recognition. Specifically, we propose a Nonlinear Compatibility Based Method (NCBM). In training phase, it learns a nonlinear compatibility function between feature space instances and semantic space class prototypes. In testing phase, each testing instance is classified to a unseen class with the highest compatibility score. This method belongs to the category of widely used compatibility based methods [2, 3, 31] in zero-shot learning. In our method, the optimization in training phase is for the whose classification process, and the model is not trained for a fixed set of unseen classes. This makes it different from previous methods in zero-shot human activity recognition.

WI '17, Leipzig, Germany

978-1-4503-4951-2/17/08…$15.00
DOI: 10.1145/3106426.3106526

Our contributions are as follows: 1. We propose a new zero-shot learning method for human activity recognition problems. 2. We define attributes for activity classes in two public datasets. These attributes form the semantic space which is an important part in zero-shot learning methods. 3. We conduct extensive experiments on three public datasets. Experimental results show that, compared with several state-of-the-art methods, our proposed method can achieve the best performance in zero-shot human activity recognition problems.

The rest of the paper is organized as follows. In Section 2 we summarize some related work. In Section 3 we give a detailed description of our method. In Section 4 we present our experiments. In Section 5 we give a conclusion.

2 RELATED WORK

This work is related to two research areas: Human Activity Recognition and Zero-Shot Learning. In this section, we briefly review related work in these two areas.

2.1 Human Activity Recognition

Human activity recognition has received lots of research in past years [5, 8, 15]. Most of existing methods are based on supervised learning algorithms [5, 8]. One limitation of these methods is that, in training phase, they require a reasonable number of labeled instances for each class to recognize. In practice, the process of sensor reading collecting and labeling is time consuming and costly. This has restricted practical applications of these methods.

To relieve the burden of instance labeling, researchers have proposed some methods making use of unlabeled training instances, such as semi-supervised learning based methods [34] and unsupervised learning based methods [16]. To solve the problem of recognizing instances belonging to previously unseen activities, [20] proposed an unseen class detection method, which can detect instances belonging to previously unseen classes. But this method cannot tell which specific class the detected instances belong to. To solve the problem of getting class labels for unseen class instances, zero-shot learning methods are needed, which are discussed in the following.

2.2 Zero-Shot Learning

Zero-shot learning has drawn much attention in recent years [2, 3, 19, 21, 31]. The motivation of zero-shot learning is to transfer information from seen classes to unseen classes via semantic space. Different semantic spaces have been utilized by existing zero-shot learning methods, such as attribute space [19], word vector space [3] and space got from text descriptions [23]. Among them, attribute space is most widely used. By attribute space, most of existing zero-shot learning methods achieve their best results.

Existing zero-shot learning methods can mainly be classified into three categories. The first category is projection based methods [10, 21]. In these methods, a projection from feature space to semantic space is learned in training phase. Testing phase consists of two steps. In the first step, instances in feature space are projected to semantic space. In the second step, projected instances are classified to unseen classes by nearest neighbor method [21], label propagation [26] or other approaches. The second category is classifier-prototype correspondence based methods [27]. In these methods, the correspondence between class prototypes in semantic space and one-vs-rest classifiers in feature space is learned in training phase. In this way, one-vs-rest classifiers for each class can be represented as a function of the corresponding class prototype. Classifiers for unseen classes can be got by taking corresponding unseen class prototypes. A large category in existing zero-shot learning methods, compatibility based methods [2, 3, 31], can also be seen as belonging to this category. Compatibility based methods learn a compatibility function between feature space instances and semantic space class prototypes. The learned compatibility function with different class prototypes can be seen as classifiers for different classes. The third category is class-relationship based methods [7, 35]. In these methods, relationships among seen and unseen classes are calculated in semantic space. This information is then transferred to feature space to help to get classifiers for unseen classes.

In human activity recognition problems, there have been some zero-shot learning methods proposed. The methods proposed by [9] and [10] belong to projection based methods. In the first step, instances in feature space are projected to semantic space by SVM classifiers [10] or a variation of CRF [9]. In the second step, classification is done in semantic space by nearest neighbor method [10] or junction tree algorithm [9]. The method proposed in [35] belongs to class-relationship based methods. In this method, relationships among seen and unseen classes are calculated in semantic space. This information is then transfered to feature space. Some pseudo instances for unseen classes are got by borrowing instances from seen classes based on the learned relationships. Classifiers for unseen classes are trained based on these pseudo instances.

3 PROPOSED METHOD

In this section, we present our proposed zero-shot learning method NCBM (Nonlinear Compatibility Based Method), which belongs to compatibility based methods.

As with other zero-shot learning methods, in our method, the input comes from two spaces: feature space and semantic space.

Input from feature space are instances got from sensor readings. When dealing with sensor readings, we adopt the widely used sliding window strategy [5] to separate sensor readings into a collection of short segments. Each segment is considered as an independent instance. As shown in Figure 1, features of each instance are extracted from sensor readings in this segment. Class label of this instance is the label that appears most times in this segment.

In semantic space, we have prototypes for each class. In this paper, we adopt binary attribute space as semantic space. It has been validated in [9] and [10] that, this space is effective in human activity recognition problems. In binary attribute space, each class prototype is a 0-1 vector. If instances of a class have an attribute, the corresponding digit in the class prototype is "1", otherwise, it is "0". As shown in Figure 1, in this space, all class prototypes can form a class-attribute matrix. In this matrix, each row corresponds to a class prototype.

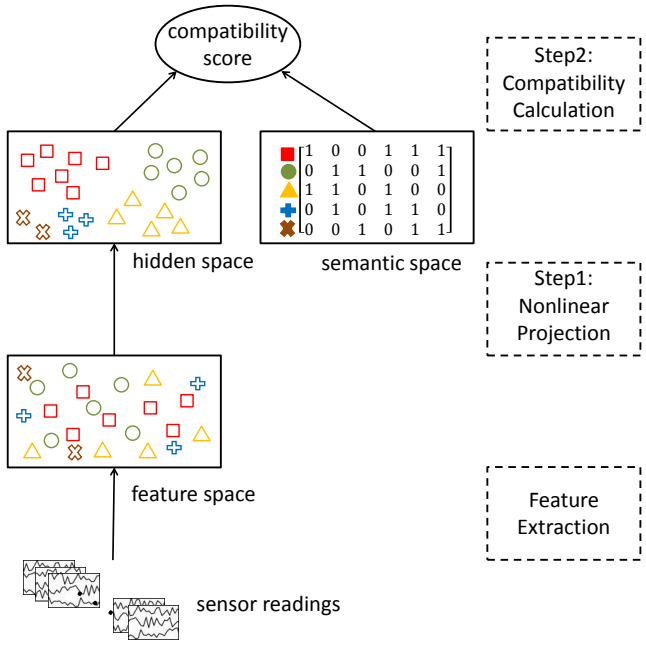

Figure 1: Process of Nonlinear Compatibility Based Method (NCBM)

3.1 Problem Formulation

The problem of zero-shot human activity recognition can be formalized as follows. In training phase, we have N^s seen classes $\mathcal{S} = \{c_i^s\}_{i=1}^{N^s}$ and N^{tr} labeled training instances $\{(\mathbf{x}_i^{tr}, y_i^{tr})\}_{i=1}^{N^{tr}} \subseteq \mathcal{X} \times \mathcal{S}$ belonging to these seen classes. $\mathcal{X} \subseteq \mathbb{R}^d$ is the feature space. Also in semantic space $\mathcal{P}$, we have class prototypes $\{\mathbf{p}_i^s\}_{i=1}^{N^s} \subseteq \{0,1\}^p$ for seen classes. In testing phase, we are given N^{ts} testing instances $\{(\mathbf{x}_i^{ts})\}_{i=1}^{N^{ts}} \subseteq \mathcal{X}$ from the same feature space. These instances belong to N^u unseen classes $\mathcal{U} = \{c_i^u\}_{i=1}^{N^u}$. Also in semantic space $\mathcal{P}$, we have class prototypes $\{\mathbf{p}_i^u\}_{i=1}^{N^u} \subseteq \{0,1\}^p$ for unseen classes. Classes covered by training and testing instances are disjoint, $\mathcal{S} \cap \mathcal{U} = \emptyset$. The goal of our method is to learned a model that can predict class labels of the testing instances.

3.2 NCBM: Proposed Nonlinear Compatibility Based Method

The motivation of our method is to learn one-vs-rest classifiers for each of the unseen classes. The classifiers can be got by the corresponding class prototypes in semantic space. This is the motivation of classifier-prototype correspondence based methods. As introduced in subsection 2.2, lot of methods in this category belong to compatibility based methods. In the following, we firstly give an introduction of compatibility based methods, then describe our proposed NCBM, finally illustrate the objective function and the way of optimization in our method.

3.2.1 Compatibility Based Methods. The motivation of compatibility based methods [1–3, 23, 31] is to learn a function measuring the degree of how "compatible" a feature space instance and a semantic space prototype is. This kind of methods follow this process: In training phase, we have labeled training instances $\{(\mathbf{x}_i^{tr}, y_i^{tr})\}_{i=1}^{N^{tr}}$ and class prototypes $\{\mathbf{p}_j^s\}_{j=1}^{N^s}$ of seen classes. With these data, we can learn a compatibility function $f(\mathbf{x}_i, \mathbf{p}_j) : \mathcal{X} \times \mathcal{P} \rightarrow \mathbb{R}$. This function takes a feature space instance $\mathbf{x}_i$ and a semantic space class prototype $\mathbf{p}_j$ as input. Output a real number indicating the degree of compatibility. In testing phase, for each testing instance, we calculate its compatibility score with all unseen class prototypes, and classify it to the unseen class with highest score.

In existing compatibility based methods, most of them are based on bilinear function [2, 3]. In these methods, the compatibility function is in form of

$$f(\mathbf{x}_i, \mathbf{p}_j) = \mathbf{x}_i^T W \mathbf{p}_j$$

which is a bilinear function takes $\mathbf{x}_i$ and $\mathbf{p}_j$ as input. $W \in \mathbb{R}^{d \times p}$ is the parameter we need to learn.

In some recent zero-shot learning methods, extension of bilinear function has been proposed. In [31], they proposed a piecewise bilinear compatibility function.

3.2.2 Proposed Nonlinear Compatibility Based Method. The compatibility function we descried above can be seen in another view. In this function, if we treat the parameter and the class prototype as a whole, we can get one-vs-rest classifiers for each class. For example, with class prototype $\mathbf{p}_j$, $f(\cdot, \mathbf{p}_j) = W\mathbf{p}_j$ is one-vs-rest classifier for instances belonging to class j. In testing phase, with prototypes of different unseen classes, we can get one-vs-rest classifiers for each of them.

By bilinear compatibility based function, all classifiers we got are linear classifiers. As an extension, by piecewise bilinear compatibility function [31], we can get piecewise linear classifiers. But in view of the classifier themselves, the classification ability of these linear and piecewise-linear classifiers is limited. In many cases, instances in feature space are not linearly separable, and also hard to be separated by piecewise-linear classifiers. For these instances, some nonlinear classifiers are need to separate them.

In category of compatibility based methods, there have been some methods proposed to learn nonlinear classifiers for unseen classes. But the nonlinear model in these methods are specific for certain data types(images [1, 4] or texts [1, 23]), cannot be used in our problem.

Also there are other methods aiming to learn classifiers for unseen classes which is a function of the corresponding class prototypes. In [12] and [33], they learn linear classes for each unseen classes. In [17] nonlinear classifiers are learned, but this method is based on the assumption that instances in each class follow Gaussian distribution.

In this paper, we propose a new method, which learns nonlinear classifiers for unseen classes. Our method belongs to compatibility based methods. We call our method Nonlinear Compatibility Based Method (NCBM). As shown in Figure 1, the process of our method can be divided into two steps.

Step 1 Nonlinear Projection. In this step, for each instance $\mathbf{x}_i$ in feature space, we project it into a hidden space $\mathcal{H} \subseteq \mathbb{R}^h$, and get a new representation $\mathbf{h}_i$ of it.

$$\mathbf{h}_i = g(\mathbf{x}_i)$$

This is achieved by a nonlinear function $g(\cdot)$.

Step 2 Compatibility Calculation. In this step, we learn a compatibility function $f(\cdot, \cdot)$ between instances in this hidden space $\mathcal{H}$ and class prototypes in semantic space $\mathcal{P}$.

$$f(\mathbf{h}_i, \mathbf{p}_j) = \mathbf{h}_i^T W \mathbf{p}_j$$

where $W \in \mathbb{R}^{h \times p}$ is the parameter.

In view of one-vs-rest classifiers, the classifiers we learned in this way are in form of $f(\cdot, \mathbf{p}_j) = g(\cdot)^T W \mathbf{p}_j$. They are nonlinear classifiers for feature space instances, and are more capable than linear classifiers.

3.2.3 Objective Function. In our method, we use hyperbolic tangent function as the nonlinear projection function.

$$g(\mathbf{x}_i) = \tanh(U\mathbf{x}_i + \mathbf{b})$$

where $U \in \mathbb{R}^{h \times d}$ is the projection matrix, $\mathbf{b} \in \mathbb{R}^h$ is the bias.

We use hinge loss as the loss function to train the model.

$$\ell = \frac{1}{N^{tr}} \frac{1}{N^s} \sum_{i=1}^{N^{tr}} \sum_{j=1}^{N^s} \max[0, \lambda - \mathbb{I}_{ij} f(g(\mathbf{x}_i), \mathbf{p}_j)]$$

where $\mathbf{x}_i$ is a feature space instance, $\mathbf{p}_j$ is a class prototype in semantic space, $g(\cdot)$ is the nonlinear projection function, $f(\cdot, \cdot)$ is the compatibility function, $\mathbb{I}_{ij}$ is an indicator that $\mathbb{I}_{ij} = 1$ if $\mathbf{x}_i$ and $\mathbf{p}_j$ come from the same class, and -1 otherwise.

3.2.4 Optimization. To minimize the loss function in our method, we use a SGD-based method Adagrad [11] to optimize the parameters. At each training epoch, from the loss function, we use the backpropagation strategy to update all of the parameters. The training process is carried to a fixed number of epochs.

4 EXPERIMENTS

4.1 Datasets and Attributes

4.1.1 Datasets and Preprocessing. To evaluate our method, we select three public datasets, which are widely used in human activity recognition problems. Each dataset can be seen as a representation of one kind of datasets got by a certain data collecting manner. PAMAP2 is the dataset got from a purely experimental setting. OPP represents the dataset got from a certain scenario. TUD is the representation of datasets got from purely daily living. Datasets PAMAP2 and OPP are got from [18].

In these datasets, for each dimension of the sensor readings, we firstly fill in missing values with 0, then normalize them into the region of $[0, 1]$.

PAMAP2[1] [24]: PAMAP2 Physical Activity Monitoring Data Set (PAMAP2) contains sensor readings of 18 activities from 9 individuals. Sampling frequency of heart rate monitor is about 9Hz, of all other sensors is 100Hz. In our experiment, we replicate the way of segmentation from the original work [24], using a sliding window of 5.12 seconds and step size of 1 second between adjacent windows. As suggested by the data provider, we discard sensor readings from 3D accelerometers with scale of ±6g (as there is another kind of 3D accelerometers, they are more accurate, and provide the same information) and orientation readings (as they are invalid in this dataset). From remaining sensor readings in each sliding window, we extract mean value and standard deviation from each dimension. After that, we get instances each having 62 dimensional features.

OPP[2] [25]: Opportunity Activity Recognition Data Set (OPP), is got from an experimental setting simulating breakfast scenario. In the process of data collecting, individuals have their freedom in sequence and manner of performing a set of activities. Sensor readings with sampling frequency of 30Hz come from 4 individuals. We use middle level activity class labels in our experiment, in which there are 17 activity classes. We follow the way of segmentation adopted by [13], using a sliding window of 1 second and step size of 0.5 second between adjacent windows. We discard data in drill runs. As they are not got by individuals freely behaving, but by letting them repeat all activities in a specific sequence. From sensor readings in each sliding window, we extract mean value and standard deviation from each dimension. After that, features of each instance are of 484 dimensions.

TUD[3] [14]: TU Darmstadt dataset (TUD) is got from daily living activities of a man. This dataset consists of sensors readings of 34 classes from two wearable accelerometers attached on him. It records everyday activities of him in a period of seven days. The original sampling frequency is 100Hz, but for the dataset that is publicly available, the sampling frequency is down sampled to 2.5Hz. We follow the way of segmentation adopted by [9, 10], using a sliding window of 30 seconds and step size of 15 seconds between adjacent windows. From the sensor readings in each sliding window, we extract mean value and standard deviation in each dimension. We also include time of day as an additional feature. After that, features of each instance are of 13 dimensions.

4.1.2 Attribute Defining for Semantic Space. In this paper, we adopt binary attribute space as semantic space. It is necessary to define attributes for each class. For TUD, we use the attributes defined in [10]. For the other two datasets, PAMAP2 and OPP, there are no defined attributes. We define attributes by ourselves.

When defining attributes, we mainly focus on two aspects: one is movement of the body, another is related object and environment. When individuals performing different activities, these activities are consisted of lots of movements. Some activities have common movements, some have different. For example, activities "running" and "walking" have the common movement "arms moving", but activity "standing" does not have. These movements are helpful to distinguish different activities. On the other hand, for some activities, the related object and environment is important for recognizing them. For example, for activity "playing soccer", "soccer" is an important object for recognizing this activity.

[1] https://archive.ics.uci.edu/ml/datasets/PAMAP2+Physical+Activity+Monitoring

[2] https://archive.ics.uci.edu/ml/datasets/OPPORTUNITY+Activity+Recognition

[3] https://www.mpi-inf.mpg.de/departments/computer-vision-and-multimodal-computing/software-and-datasets/

Table 1: Statistics of Number of Instances and Classes Belonging to Seen and Unseen Classes in Each Fold in Three Datasets

Fold	Number of Instances						Number of Classes					
	PAMAP2		OPP		TUD		PAMAP2		OPP		TUD	
	seen classes	unseen classes	seen classes	unseen classes	seen classes	unseen classes	seen classes	unseen classes	seen classes	unseen classes	seen classes	unseen classes
fold 1	21461	5781	6007	1164	16423	646	14	4	13	4	26	7
fold 2	21465	5777	5617	1554	16704	365	14	4	13	4	26	7
fold 3	19995	7247	5062	2109	3619	13450	14	4	14	3	26	7
fold 4	22736	4506	6148	1023	16460	609	15	3	14	3	27	6
fold 5	23311	3931	5850	1321	15070	1999	15	3	14	3	27	6

When we define attributes, we also refer to attributes defined in [10] and [19]. We define 42 attributes for PAMAP2, and 15 attributes for OPP. The detail of defined attributes for each class can be seen in Appendix B.

In these three datasets, there are lots of instances belonging to null class. Null class refers to instances belonging to either not relevant activities or no activities at all [32]. For this class, it is not meaningful to define attributes for it. In our experiment, we discard instances belonging to this class.

In attributes defined by [10] for TUD, there are no attributes for activity "preparing food". In our experiment, we also discard instances belonging to this class.

4.2 Experimental Setup

4.2.1 Compared Methods. To evaluate our method, we compare our method with 4 baseline methods coming from different categories. In zero-shot human activity recognition problems, there are no works in categories of classifier-prototype correspondence based method and class-relationship based method utilizing attribute space as semantic space. So in our experiment, we use state-of-the-art methods in other zero-shot learning problems belonging to these two categories.[4]

RegBM [21]: a projection-based method. The projection is achieved by a regularized multiple output linear regression model. In this paper, we call this method Regression-based method (RegBM).

SVMBM [10]: a projection-based method. The projection is achieved by SVM classifiers for each dimension in semantic space. In this paper, we call this method SVM-classifier-based method (SVMBM). This method is a part of the activity recognition system NuActiv in [10].

BCBM: bilinear compatibility based method (BCBM). A classifier-prototype correspondence based method. It is the degeneration of our method.

SynC [7]: a class-relationship based method. In [7] they used three different loss functions to train the model, we compare with all these three implementations.

4.2.2 Implementation Details. Our method is implemented by Theano [29]. In training phase, we use mini-batch of size 100. In each training epoch, we use Adagrad [11] to update the parameters.

[4]For RegBM and SVMBM, there are no source code publicly available, so we implement them by ourselves. For SynC, we use the source code provided by the authors. For bilinear compatibility based methods, there are different implementations in different works. In our experiment, except the model, we use the same implementation of our NCBM. In this way, the comparison of experimental results are more persuasive.

Table 2: Comparison on Average Per-Class Accuracy (in %) of Different Methods

Method	PAMAP2	OPP	TUD
RegBM*	53.16	55.62	28.99
SVMBM	49.99	27.60	29.53
BCBM	60.61	76.31	30.19
SynC-ovso	47.24	68.62	26.01
SynC-cs	50.82	75.74	36.81
SynC-struct	54.42	78.55	31.90
NCBM**	64.37	84.10	37.35

* in this method, the results in PAMAP2 and OPP are got when $\lambda = 0.1$, in TUD when $\lambda = 0.01$.

** in this method, the results in PAMAP2 and TUD are got when hidden space of dimension 80, in OPP when hidden space of dimension 90.

For PAMAP2 and OPP, we set the initial learning rate to be 0.01. For TUD, we set the initial learning rate to be 0.5. In all of these three datasets, we set the number of training epochs to be 2000, the value of λ in the loss function to be 1. In our experiment, we evaluate the impact of different hidden space dimensions. We will give a detailed description of it in subsection 4.4.

4.2.3 Evaluation Metrics. In our experiment, we use average per-class accuracy as evaluation metric. It is the most widely used metric to evaluate zero-shot learning methods in existing literatures [3, 7, 23, 27, 31]. Average per-class accuracy is the average of accuracy value of each class. In activity recognition problems, the number of instances belonging to different classes differs a lot. If accuracy is averaged over all testing instances, the test result will inclines to classes with more instances. Average per-class accuracy is a metric does not influenced by this phenomenon. It is suitable for our setting.

4.2.4 Zero-Shot Setting Design. In our experiment, we use cross validation to evaluate the effectiveness of our method. We randomly split classes in each dataset into 5 disjoint folds, and take classes in each fold as unseen classes in turn. Table 1 is the statistics of number of instances and classes belonging to seen and unseen classes in each fold. Detailed information of class splitting is in Appendix A.

4.3 Evaluation on Average Per-Class Accuracy

Table 2 is the results of different methods on these three datasets. The value in each table cell is average results of the 5 folds.

From the experiment results, we can see classifier-prototype correspondence based methods and class-relationship based methods outperform projection based methods in most cases.

For RegBM, we test the regularization parameter λ in range of {0.01, 0.1, 1, 10, 100}, and report the best results on each dataset. In projection based methods, RegBM outperforms SVMBM (only on TUD, they get similar results). This is because in RegBM, it learns a projection function which takes all dimensions of the semantic space as a whole, the projection is optimized for all dimensions. While in SVMBM, classifiers for each dimension are trained independently. In this way, the optimization for each classifier is not for the whole projection. As a result, the performance of SVMBM is not good in general.

In general, the performances of BCBM and SynC are comparable. Although on a specific dataset, one method may outperform another, but in general, their performances are not of much difference. They are representations of classifier-prototype correspondence based methods and class-relationship based methods respectively. This illustrates that the performances of existing state-of-the-art methods in these two categories are similar.

In category of classifier-prototype correspondence based methods, NCBM outperforms BCBM. This agrees with our perceive that nonlinear model is more capable than bilinear model.

In all of these three datasets, our method NCBM achieves state-of-the-art results. This illustrates that our proposed method is suitable for zero-shot human activity recognition problem and can get state-of-the-art results in this problem.

4.4 Impact of Hidden Space Dimension

We evaluate the impact of dimension of the hidden space. The results are shown in Figure 2.

In the experiment, we change the hidden space dimension from 10 to 100, with interval of 10. Figure 2 shows the average results of the 5 folds at different dimensions. From the figure we can see, when the hidden space dimension is extremely low (in case of 10), the performances on all these three datasets are relatively poor. With the increase of the hidden space dimension, the performances get better, finally the performances become stable.

When the hidden space dimension is low, the capability of the model in our method is limited. With the increase of the dimension, the capability increases. This phenomenon also illustrates the necessity of nonlinear model. Nonlinear classifiers are more capable than linear classifiers. In zero-shot learning problems, it can achieve better performances. When the dimension of hidden space is within a reasonable scale (in scale of 50 to 100), the performances on these three datasets are relatively stable. We can conclude that, within a reasonable hidden space dimension scale, our method is not sensitive to the dimension of hidden space.

4.5 Evaluation on Training Time

All the experiments are conducted on a computer with Intel Core i7 CPU and 16GB memory. Table 3 is the statistics of time spent by different methods in training model. The value in each table cell is the average of time in the 5 folds.

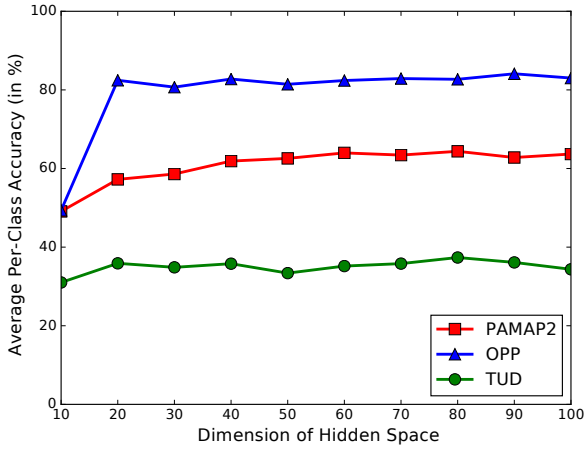

Figure 2: Average Per-Class Accuracy on Three Datasets with Different Hidden Space Dimensions

Table 3: Comparison on Model Training Time (in seconds) of Different Methods

Method	PAMAP2	OPP	TUD
RegBM	-	-	-
SVMBM	886.97	89.92	18.87
BCBM	213.90	48.93	90.88
SynC-ovso	158.54	172.19	99.31
SynC-cs	313.95	515.82	236.55
SynC-struct	312.61	511.00	237.19
NCBM*	347.99	159.55	201.37

* in this method, the training time in PAMAP2 and TUD is when hidden space of dimension 80, in OPP is when hidden space of dimension 90.

In RegBM, it uses a multi output linear regression model. In this model, analytical solutions can be got for all of the parameters. So in this method, there is not an obvious process of model training.

In SVMBM, it trains SVM classifiers for each dimension in semantic space. So the training time of it is sensitive to the dimension of semantic space. In PAMAP2, the dimension of semantic space is 42, much larger than dimension of semantic spaces in OPP (which is 15) and TUD (which is 17). So in PAMAP2, SVMBM needs extreme long time for model training. From the view of model training time, we can conclude that SVMBM is not scalable to high dimensional semantic spaces.

In SynC, the three implementations take different time to train the model. This is mainly because of different loss functions used in different implementations.

Compared with BCBM, our NCBM takes longer time for model training. It is consistent with our perceive. In NCBM, there is an additional projection step, and more parameters to learn. In TUD, the dimension of both feature space and semantic space are not high, but the training time of our NCBM is not very short. This is probability because on TUD, we set the initial learning rate to be

Table 4: Class Splitting in PAMAP2

Fold	Classes
fold 1	watching TV, house cleaning, standing, ascending stairs
fold 2	walking, rope jumping, sitting, descending stairs
fold 3	playing soccer, lying, vacuum cleaning, computer work
fold 4	cycling, running, Nordic walking
fold 5	ironing, car driving, folding laundry

Table 5: Class Splitting in OPP

Fold	Classes
fold 1	close drawer 2, clean table, toggle switch, open drawer 3
fold 2	open fridge, open door 2, close drawer 1, open drawer 2
fold 3	drink from cup, open drawer 1, close dishwasher
fold 4	close drawer 3, close door 2, open door 1
fold 5	close fridge, open dishwasher, close door 1

0.5. In our implementation, if the learning rate is set to be relatively large, the time of model training will be relatively long.

From Table 3, we can see, compared with other methods, the training time of our method is acceptable, also our method can get the best performances. In view of the time spend in model training, our method is also a suitable method.

5 CONCLUSION

In this paper, we propose a novel nonlinear compatibility based zero-shot learning method for human activity recognition problems. The proposed method is not only suitable for activity recognition, but can also be generated to other zero-shot learning problems. We investigate the performance of the proposed method on three public datasets. And compare it with state-of-the-art methods. Experimental results show that, compared with state-of-the-art methods, our method can achieve the best performance in human activity recognition problems. Also our model takes acceptable time in model training and within a reasonable dimension scale, our model is not sensitive to the dimension of hidden space.

In the future, we plan to explore semi-supervised zero-shot learning methods for human activity recognition. As in human activity recognition problems, labeled instances are usually limited and labeling process is costly and time consuming. Besides, we are also interested in investigating the performance of the proposed method on the other datasets, such as images and videos.

A SPLITTING OF CLASSES

Table 4 is the splitting of classes in PAMAP2. Table 5 is the splitting of classes in OPP. Table 6 is the splitting of classes in TUD.

Table 6: Class Splitting in TUD

Fold	Classes
fold 1	making coffee, standing / talking, attending a presentation, discussing at whiteboard, fanning barbecue, running, walking while carrying something
fold 2	standing / talking on phone, standing / having a coffee, walking, wiping the whiteboard, queuing in line, sitting / talking on phone, brushing teeth
fold 3	driving bike, having breakfast, kneeling / doing something else, driving car, watching a movie, sitting / desk activities, standing / using the toilet
fold 4	washing hands, picking up cafeteria food, having lunch, personal hygiene, using the toilet, washing dishes
fold 5	walking freely, kneeling / making fire for barbecue, setting the table, lying while reading / using computer, sitting / having a coffee, having dinner

Table 7: Defined Attributes for OPP

Attribute Aspect	movement of body					related object and environment								
Activity \ Attribute	open	close	clean	drink	toggle	door 1	door 2	fridge	dishwasher	drawer 1	drawer 2	Drawer 3	table	cup	switch
open door 1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
open door 2	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0
close door 1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0
close door 2	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0
open fridge	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
close fridge	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0
open dishwasher	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
close dishwasher	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0
open drawer 1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
close drawer 1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
open drawer 2	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0
close drawer 2	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
open drawer 3	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0
close drawer 3	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0
clean table	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0
drink from cup	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
toggle switch	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1

B DEFINED ATTRIBUTES FOR PAMAP2 AND OPP

Table 7 is defined attributes for OPP. Table 8 is defined attributes for PAMAP2.

ACKNOWLEDGMENTS

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its IDM Futures Funding Initiative; and the Interdisciplinary Graduate School (IGS), NTU.

REFERENCES

[1] Zeynep Akata, Mateusz Malinowski, Mario Fritz, and Bernt Schiele. 2016. Multi-cue zero-shot learning with strong supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 59–68.

[2] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. 2013. Label-embedding for attribute-based classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 819–826.

Table 8: Defined Attributes for PAMAP2

Attribute Aspect	movement of body																																related object and environment													
Attribute / Activity	motion	static	cyclic motion	intense motion	translation motion	free motion	body vertical	body incline	body horizontal	body forward	body backward	body up	body down	body in place	torso transform	arms motion	arms static	arms bent	arms straight	arms bent-straight transform	hands hold something	legs motion	legs static	legs bent	legs straight	legs bent-straight transform	legs alternate move forward	legs move up and/or down	seat	bike	poles	television	computer	car	stairs	vacuum	iron	clothes	soccer	rope	indoor	outdoor
lying	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	1	1	1	1	1	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
sitting	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	1	1	1	1	1	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1
standing	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	1	1	1	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
walking	1	0	1	0	1	0	1	0	0	1	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
running	1	0	1	1	1	0	1	0	0	1	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
cycling	1	0	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	1	1	1	0	1	1	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1
Nordic walking	1	0	1	0	1	0	1	0	0	1	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
watching TV	0	1	0	0	0	0	1	1	0	0	0	0	0	1	0	1	1	1	1	1	0	1	1	1	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0
computer work	0	1	0	0	0	0	1	1	0	0	0	0	0	1	0	1	1	1	1	1	1	1	1	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0
car driving	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	1	0	0	1	1	1	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1
ascending stairs	1	0	1	0	0	0	1	1	0	1	0	1	0	0	0	1	0	1	1	1	1	0	0	0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1
descending stairs	1	0	1	0	0	0	1	1	0	1	0	0	1	0	0	1	0	1	1	1	1	1	0	0	0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	1	1
vacuum cleaning	1	0	0	0	0	1	1	1	0	1	1	0	0	0	1	1	0	1	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0
ironing	1	0	0	0	0	1	1	1	0	0	0	0	0	0	1	1	1	0	1	1	1	1	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0	1	0
folding laundry	1	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0
house cleaning	1	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	0	1	1	0	0	0	0	0	0	1	0	0	0	0	1	0
playing soccer	1	0	0	1	0	1	1	1	0	1	0	1	1	0	1	1	0	1	1	1	0	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
rope jumping	1	0	1	1	0	0	1	0	0	0	0	1	1	1	0	1	0	1	1	1	1	1	0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1

[3] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. 2015. Evaluation of output embeddings for fine-grained image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2927–2936.

[4] Lei Jimmy Ba, Kevin Swersky, Sanja Fidler, and others. 2015. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *Proceedings of the IEEE International Conference on Computer Vision*. 4247–4255.

[5] Andreas Bulling, Ulf Blanke, and Bernt Schiele. 2014. A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys (CSUR)* 46, 3 (2014), 33.

[6] José Carlos Castillo, Davide Carneiro, Juan Serrano-Cuerda, Paulo Novais, Antonio Fernández-Caballero, and José Neves. 2014. A multi-modal approach for activity classification and fall detection. *International Journal of Systems Science* 45, 4 (2014), 810–824.

[7] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. 2016. Synthesized classifiers for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5327–5336.

[8] Liming Chen, Jesse Hoey, Chris D Nugent, Diane J Cook, and Zhiwen Yu. 2012. Sensor-based activity recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42, 6 (2012), 790–808.

[9] Heng-Tze Cheng, Martin Griss, Paul Davis, Jianguo Li, and Di You. 2013. Towards zero-shot learning for human activity recognition using semantic attribute sequence model. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. ACM, 355–358.

[10] Heng-Tze Cheng, Feng-Tso Sun, Martin Griss, Paul Davis, Jianguo Li, and Di You. 2013. Nuactiv: Recognizing unseen new activities using semantic attribute-based learning. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. ACM, 361–374.

[11] John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research* 12, Jul (2011), 2121–2159.

[12] Mohamed Elhoseiny, Babak Saleh, and Ahmed Elgammal. 2013. Write a classifier: Zero-shot learning using purely textual descriptions. In *Proceedings of the IEEE International Conference on Computer Vision*. 2584–2591.

[13] Nils Y. Hammerla, Shane Halloran, and Thomas Plötz. 2016. Deep, Convolutional, and Recurrent Models for Human Activity Recognition Using Wearables. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. 1533–1540.

[14] Tâm Huynh, Mario Fritz, and Bernt Schiele. 2008. Discovery of activity patterns using topic models. In *Proceedings of the 10th international conference on Ubiquitous computing*. ACM, 10–19.

[15] Eunju Kim, Sumi Helal, and Diane Cook. 2010. Human activity recognition and pattern discovery. *IEEE Pervasive Computing* 9, 1 (2010), 48–53.

[16] Yongjin Kwon, Kyuchang Kang, and Changseok Bae. 2014. Unsupervised learning for human activity recognition using smartphone sensors. *Expert Systems with Applications* 41, 14 (2014), 6067–6074.

[17] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. 2008. Zero-data Learning of New Tasks. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*. 646–651.

[18] M. Lichman. 2013. UCI Machine Learning Repository. (2013). http://archive.ics.uci.edu/ml

[19] Jingen Liu, Benjamin Kuipers, and Silvio Savarese. 2011. Recognizing human actions by attributes. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 3337–3344.

[20] Le T Nguyen, Ming Zeng, Patrick Tague, and Joy Zhang. 2015. I did not smoke 100 cigarettes today!: avoiding false positives in real-world activity recognition. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 1053–1063.

[21] Mark Palatucci, Dean Pomerleau, Geoffrey E Hinton, and Tom M Mitchell. 2009. Zero-shot learning with semantic output codes. In *Advances in neural information processing systems*. 1410–1418.

[22] Vijay Rajanna, Raniero Lara-Garduno, Dev Jyoti Behera, Karthic Madanagopal, Daniel Goldberg, and Tracy Hammond. 2014. Step up life: a context aware health assistant. In *Proceedings of the Third ACM SIGSPATIAL International Workshop on the Use of GIS in Public Health*. ACM, 21–30.

[23] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. 2016. Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 49–58.

[24] Attila Reiss and Didier Stricker. 2012. Introducing a new benchmarked dataset for activity monitoring. In *2012 16th International Symposium on Wearable Computers*. IEEE, 108–109.

[25] Daniel Roggen, Alberto Calatroni, Mirco Rossi, Thomas Holleczek, Kilian Förster, Gerhard Tröster, Paul Lukowicz, David Bannach, Gerald Pirkl, Alois Ferscha, and others. 2010. Collecting complex activity datasets in highly rich networked sensor environments. In *Seventh International Conference on Networked Sensing Systems*. IEEE, 233–240.

[26] Marcus Rohrbach, Sandra Ebert, and Bernt Schiele. 2013. Transfer learning in a transductive setting. In *Advances in neural information processing systems*. 46–54.

[27] Bernardino Romera-Paredes and Philip Torr. 2015. An embarrassingly simple approach to zero-shot learning. In *Proceedings of the 32nd International Conference on Machine Learning*. 2152–2161.

[28] Nagender Kumar Suryadevara and Subhas Chandra Mukhopadhyay. 2012. Wireless sensor network based home monitoring system for wellness determination of elderly. *IEEE Sensors Journal* 12, 6 (2012), 1965–1972.

[29] Theano Development Team. 2016. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints* abs/1605.02688 (May 2016). http://arxiv.org/abs/1605.02688

[30] Jie Wan, Michael J O'grady, and Gregory M O'hare. 2015. Dynamic sensor event segmentation for real-time activity recognition in a smart home context. *Personal and Ubiquitous Computing* 19, 2 (2015), 287–301.

[31] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. 2016. Latent embeddings for zero-shot classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 69–77.

[32] Jianbo Yang, Minh Nhut Nguyen, Phyo Phyo San, Xiaoli Li, and Shonali Krishnaswamy. 2015. Deep Convolutional Neural Networks on Multichannel Time

Series for Human Activity Recognition. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*. 3995–4001.

[33] Yongxin Yang and Timothy M Hospedales. 2014. A unified perspective on multi-domain and multi-task learning. In *International Conference on Learning Representations*.

[34] Lina Yao, Feiping Nie, Quan Z Sheng, Tao Gu, Xue Li, and Sen Wang. 2016. Learning from less for better: semi-supervised activity recognition via shared structure discovery. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 13–24.

[35] Vincent Wenchen Zheng, Derek Hao Hu, and Qiang Yang. 2009. Cross-domain activity recognition. In *Proceedings of the 11th international conference on Ubiquitous computing*. ACM, 61–70.