AttriNet: learning mid-level features for human activity recognition with deep belief networks

Conference Paper · September 2019

DOI: 10.1145/3341162.3345600

Shunwen Tan
University of California, Los Angeles
1 PUBLICATION **1** CITATION
SEE PROFILE

Ming Zeng
Carnegie Mellon University
22 PUBLICATIONS **452** CITATIONS
SEE PROFILE

Ole J. Mengshoel
Carnegie Mellon University
131 PUBLICATIONS **1,646** CITATIONS
SEE PROFILE

John Shen
Carnegie Mellon University
187 PUBLICATIONS **6,742** CITATIONS
SEE PROFILE

AttriNet: Learning Mid-Level Features for Human Activity Recognition with Deep Belief Networks

Harideep Nair
harideep.nair@sv.cmu.edu
Carnegie Mellon University

Cathy Tan
shunwen.tan.2017@anderson.ucla.edu
University of California, Los Angeles

Ming Zeng
ming.zeng@sv.cmu.edu
Carnegie Mellon University

Ole J. Mengshoel
ole.mengshoel@sv.cmu.edu
Carnegie Mellon University,
Norwegian University of Science and Technology

John Paul Shen
john.shen@sv.cmu.edu
Carnegie Mellon University

ABSTRACT

Human activity recognition (HAR) is essential to many context-aware applications in mobile and ubiquitous computing. A human's physical activity can be decomposed into a sequence of simple actions or body movements, corresponding to what we denote as mid-level features. Such mid-level features ("leg up," 'leg down," "leg still," ...), which we contrast to high-level activities ("walking," "sitting," ...) and low-level features (raw sensor readings), can be developed manually. While proven to be effective, this manual approach is not scalable and relies heavily on human domain expertise. In this paper, we address this limitation by proposing a machine learning method, AttriNet, based on deep belief networks. Our AttriNet method automatically constructs mid-level features and outperforms baseline approaches. Interestingly, we show in experiments that some of the features learned by AttriNet highly correlate with manually defined features. This result demonstrates the potential of using deep learning techniques for learning mid-level features that are semantically meaningful, as a replacement to handcrafted features. Generally, this empirical finding provides an improved understanding of deep learning methods for HAR.

CCS CONCEPTS

• **Computing methodologies → Artificial intelligence**; **Activity recognition and understanding**.

KEYWORDS

Activity Recognition, Deep Belief Network, Semantic Feature

ACM Reference Format:
Harideep Nair, Cathy Tan, Ming Zeng, Ole J. Mengshoel, and John Paul Shen. 2019. AttriNet: Learning Mid-Level Features for Human Activity Recognition with Deep Belief Networks. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2019 International Symposium on Wearable Computers (UbiComp/ISWC '19 Adjunct), September 9–13, 2019, London, United Kingdom.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3341162.3345600

1 INTRODUCTION

Human activity recognition (HAR) through smartphones has been an indispensable component in mobile ubiquitous computing. As a foundation, HAR enables many context-aware applications and services [9, 29, 30]. To recognize activities of a mobile user, various machine learning (ML) algorithms have been applied and engineered for specific application contexts [2, 16].

Many existing ML methods use labeled training data for every single activity class that the HAR system aims to detect. However, this methodology omits some useful information. For example, rich structural information of the "chest press" activity as shown in Figure 1 can hardly be characterized by such a single class label. At the same time, most existing approaches have to enumerate all existing activity classes, and cannot recognize a previously unseen activity if there were no training samples for that activity [8]. One popular solution to these challenges is to introduce mid-level features that capture higher level concepts [8, 16]. One approach to introduce such mid-level features is by manually designing semantic attributes [7, 8]. This approach has also proven effective in computer vision [11, 20, 21]. Researchers have also applied it in HAR and achieved satisfactory results [7, 8].

Figure 1 illustrates the attribute concept in activity recognition. The workout activity "chest press" may be effectively

UbiComp/ISWC '19 Adjunct, September 9–13, 2019, London, United Kingdom

ACM ISBN 978-1-4503-6869-8/19/09…$15.00
https://doi.org/10.1145/3341162.3345600

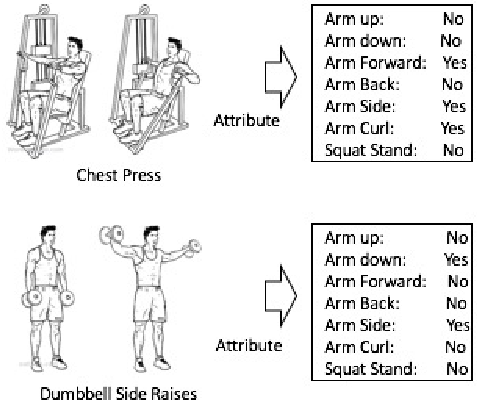

Figure 1: High-level activities can be represented by a set of mid-level attributes.

represented by introducing a set of mid-level attributes: "arm forward," "arm side," "arm curl," and so forth.

The attribute representation can be obtained through the following steps: (1) an expert with domain knowledge defines a set of attributes, and each instance in the training dataset has to be labeled with the presence or absence of each attribute; (2) a classifier is trained for each of the attributes using the training data; (3) a feature selection scheme is applied on the attributes to create appropriate feature combination [11]. However, obtaining these attributes is often time consuming and expensive since it requires much effort from test subjects, human annotators and domain experts. This demanding procedure also suffers from a scalability issue when new activities and new low-level features are present. Moreover, selecting attributes manually can be subjective and arbitrary, and may lead to non-discriminative features.

Some unsupervised learning algorithms attempt to construct mid-level features instead of attributes. Some approaches rely on latent Dirichlet allocation (LDA) [5], which uses a set of topics to describe activities. LDA has been successfully applied in text analysis, information retrieval, computer vision [17], and human activity recognition [16]. However, unlike words in text, activity signals have less clear semantic interpretations. Therefore, LDA has not been very successful in identifying mid-level feature representations.

Another line of work is represented by deep neural networks, which learn a hierarchical set of features in an unsupervised or supervised manner. For example, the idea underlying Deep Belief Network (DBN) is to use restricted Boltzmann machine (RBM) [14] as a building block. This enables the use of a greedy layer-wise learning procedure. RBM is a bi-partite undirected graphical model that is capable of learning a dictionary of patterns. These patterns are positively correlated with the observed input data. In computer vision, DBNs have achieved promising results [21]. Although deep learning approaches have been applied in activity recognition task [25, 32, 33, 35], the area is under-researched [18]. In this paper, we expand the RBM into a hierarchical representation, wherein relevant semantic concepts are revealed at the higher levels. Additionally, we use Indian buffet process (IBP) to train a sparse DBN, which helps to get more relevant semantic features and improve the results.

In order to identify the semantic concepts that are captured by the mid-level features by a sparse DBN, we carry out experiments and evaluate the performance. By computing the correlation between learned features and each of the labeled attributes in the training set, we can evaluate the correspondences between the learned features and the labeled attributes. We demonstrate that we can find semantic concepts similar to attributes like "arm up" and "arm down," even though no information with regards to these attributes was given during the training process. Improved accuracy further demonstrates that HAR applications can benefit from deep learning approaches.

We summarize our key contributions as follows:

- We propose an approach that uses a heterogeneous sparse DBN to extract mid-level feature representation without using any domain knowledge.
- We also demonstrate that learned features carry appropriate semantic meaning by calculating and evaluating correlation with available manually defined attributes.

The paper is organized as follows. We begin with a survey of related work and discuss how it compares to our work. Next, we present our approach built on the restricted Boltzmann machine and Deep Belief Networks (DBNs). Furthermore, we propose a sparse DBN-based mechanism that enhances the results. We thereafter present experimental results and analysis. Finally, we conclude and discuss future research directions.

2 RELATED WORK

In the field of mobile, wearable, and pervasive computing, extensive research has been conducted to recognize human activities [2, 4, 7, 8, 23–25, 31–35]. One line of research in this field starts with Bao et al. [2], who placed accelerometers on different body positions to recognize daily activities such as "walking," "sitting," and "watching TV." Since then, researchers have been devoted to improving recognition accuracy. Many of them investigated underlying structural representations of activities. For example, Peng el at. [24] apply the hidden Markov model (HMM) to model activities using one latent layer.

The idea of latent structure was extended for recognizing previously unseen activities. Cheng et al. [7, 8] leverage zero-shot learning [22] in the NuActiv approach, using predefined semantic mid-level attributes to predict new activities. Essentially, the manually defined attributes can be regarded as mid-level features. The introduction of such features have been proven effective in computer vision, for instance in object recognition [17, 20, 26].

Manually defining attributes, however, is time-consuming and expensive. To address these drawbacks, Mittelman et al. [21] propose the Beta-Bernoulli process restricted Boltzmann machine (BBP-RBM) to learn mid-level features for object recognition. In HAR, there are similar approaches attempting to construct mid-level features using latent Dirichlet allocation (LDA) [16]. Huynh et al. showed that LDA-based approaches, however, are limited to features that have high correlation with the activities to be recognized [16]. Deep neural networks represent another line of study to learn hierarchical features in an unsupervised manner. Thomas et al. [25] applied the RBM to extract features from accelerometer data. Zeng et al. [33] took advantage of convolutional neural network to preserve local dependency and scale invariant features to achieve better recognition performance. In contrast, we are, in this paper, able to leverage a DBN to learn relevant mid-level semantic features pertaining to HAR without requiring manually defined attributes.

To avoid overfitting in training, sparsity is introduced into deep neural networks [13, 19, 27, 28]. Advantages of sparsity also include information disentangling and efficient variable-size representation [13]. One popular sparsity technique is dropout [28], which randomly removes some nodes in each iteration during the training procedure. Lee et al. [19] set thresholds in the node selection phase of RBM to enforce sparsity penalty. Mittelman et al. [21] use a Beta-Bernoulli process over the RBM to remove some nodes. Sourav et al. [3] use a sparse-coding framework to build a feature space codebook onto which the transportation activities in their experiment were mapped. In this work, we also introduce heterogeneous sparsity into our DBN in order to achieve superior results.

3 DBN WITH HETEROGENEOUS SPARSITY FOR LEARNING MID-LEVEL FEATURES

Standard Restricted Boltzmann Machine

An RBM is a two-layer undirected probabilistic graph, in which the visible input layer contains a set of binary or real valued units $\{v_1, ..., v_{N_v}\}$ and the hidden layer is composed of a set of binary units $\{h_1, ..., h_{N_h}\}$. Here, N_v and N_h are the numbers of visible units and hidden units respectively. Connections are only allowed between the visible layer and the hidden layer. Let $v = [v_1, ..., v_{N_v}]^T$ and $h = [h_1, ..., h_{N_h}]^T$, where T denotes the transpose. The energy function of RBM is defined as

$$E(v, h) = -h^T W v - b^T v - c^T h \tag{1}$$

where $W = [w_{ji}]_{N_h \times N_v}$ is the weight matrix, $b = [b_i]_{N_v \times 1}$ is the bias of visible units and $c = [c_j]N_h \times 1$ is the bias of hidden units. Then the joint probability distribution of v and h with σ as the activation function is

$$p(h_k|v) = \sigma(w_{k,i}v_i + b_k) \tag{2}$$

$$p(v_i|h) = \sigma(w_{k,i}h_k, +c_i) \tag{3}$$

The log likelihood function corresponding to the visible units is given by

$$P(v) = \frac{1}{Z}\sum_h (-E(v, h)) \tag{4}$$

where Z is the normalization factor.

We denote the parameters of RBM by $\theta = \{W, b, c\}$. The derivative of the log-likelihood of visible units (P(v)) with respect to model parameter θ can be written as

$$\frac{\partial P(v)}{\partial \theta} = \mathbb{E}_{data}\left(-\frac{\partial E(v, h)}{\partial \theta}\right) - \mathbb{E}_{model}\left(-\frac{\partial E(v, h)}{\partial \theta}\right) \tag{5}$$

where $E_{data}(\cdot)$ and $E_{model}(\cdot)$ denote the expectations of the data distribution and the model distribution, respectively. Computing the function $\frac{\partial P(v)}{\partial \theta}$ in (5) exactly is intractable because the closed form of the model distribution remains unknown. However, the derivative can be approximately computed by Contrastive Divergence (CD) [14]. With CD, the locally optimal solutions of model parameters θ can be attained by gradient descents.

RBM with Random Dropout Sparsity

Dropout training controls overfitting by randomly omitting subsets of features at each iteration of a training procedure [15]. Formally, we can use $F = f_1, ..., f_K$, to represent an indicator vector, $F \in 0, 1$. Each f_k is generated according to a uniform distribution, $f_k \sim U(\gamma)$. In each iteration, F is enforced on each input layer to remove nodes using f_k.

Indian Buffet Process

The Indian buffet process (IBP) can be applied to generate a binary indicator vector with similar 0/1 patterns. It is natural to combine with the RBM probability model. We use $\mathbf{z}_{IBP} = [z_1, ..., z_K]$ to denote the indicator vector. We assume the two-parameter IBP [12], and use $Z \sim IBP(\alpha, \beta)$to indicate the vector $Z_{IBP} \in \{0, 1\}^K$. Specifically, the indicator vector Z is generated according to a Beta-Bernoulli process as follows:

$$\begin{aligned} \pi &\sim Beta(\alpha/K, \beta(K-1)/K), \\ z_k &\sim Bernoulli(\pi_k) \end{aligned} \tag{6}$$

where α, β are positive parameters, and we use the notation $\pi = [\pi_1, ...\pi_K]^T$ for the parameters of the Bernoulli distribution. It is implied from (6) that if π_k is close to 1 then z_k is more likely to be 1, and vice versa. The form of the parameters of Beta distribution implies that for a sufficiently large K and a reasonable choice of α and β, most π_k will be close to zero, which implies a sparsity constraint on z_k.

RBM with IBP Sparsity

In this section, we enhance the generalization ability of RBM from a different perspective - by enforcing constraints on the nodes of hidden layer. Dropout increases sparsity by removing hidden nodes uniformly in each training epoch. However, by leveraging IBP, we demonstrate we are able to obtain better sparse features due to IBP's grouping characteristic [1].

The binary selection vector $z = [z_1, ..., z_K]^T$ is used to choose which of the K hidden units should be allowed to remain activated. Our approach is to define an undirected graphical model in the form of a factor graphical model. Using the binary selection vector mentioned above, we have a new energy function

$$E(v, h, z) = -(z \otimes h)^T Wv - b^T (z \otimes h) - c^T v, \qquad (7)$$

where $\otimes$ denotes element-wise vector multiplication. With the new energy function, we can define

$$g_1(v, h, z) = e^{-E(v,h,z)} \qquad (8)$$

Since the binary selection vector is created via Beta-Bernoulli Process, its distribution function can be described as

$$g_2\left(\left\{z^j\right\}_{j=1}^{M}, \pi\right) = \prod_{k=1}^{K} \pi_k^{\sum_{j=1}^{M} z_k^j} (1 - \pi_k)^{\sum_{j=1}^{M}(1-z_k^j)} \times \pi_k^{\alpha/K-1} (1 - \pi_k)^{\beta(K-1)/K-1} \qquad (9)$$

where j denotes the index of the training sample, and M represents the number of training samples.

Using the training factor graph, the PDF for IBP-RBM is

$$p\left(\left\{v^j, h^j, z^j\right\}_{j=1}^{M}, \pi\right) \propto \prod_{j=1}^{M} g_1\left(v^j, h^j, z^j\right) g_2\left(\left\{z^j\right\}_{j=1}^{M}, \pi\right) \qquad (10)$$

IBP-RBM Inference

Inference in IBP-RBM can be estimated by Gibbs sampling. The joint posterior PDF of h and z can be sampled as below

$$p(h_k = a, z_k = b | v_k, \pi_k) \propto \begin{cases} \pi_k e^{\sum_i w_{k,i} v_i} & a = 1, b = 1 \\ \pi_k & a = 0, b = 1 \\ 1 - \pi_k & a = 0, b = 0 \\ 1 - \pi_k & a = 1, b = 0 \end{cases} \qquad (11)$$

Then the posterior PDF of π takes the form

$$\pi_k \sim Beta\left(\alpha/K + \sum_{j=1}^{M} z_k, \beta(K-1)/K + \sum_{j=1}^{M}(1 - z_k)\right) \qquad (12)$$

Sampling from the posterior PDF of the visible layer is performed in a similar manner as described in standard RBM.

DBN with Heterogeneous Sparsity

Once a layer of the network is trained, the parameters w_{ij}, b_j, c_i's are frozen and the hidden unit values are inferred from the given data. These inferred values act as the "data" that will be used to train the next higher layer in the network. We use dropout on the first hidden layer and use IBP on the second hidden layer, which injects heterogeneous sparsity to the DBN (HSparseDBN). Fig. 2 shows the structure of the HSparseDBN model. The details of our procedure are summarized in Algorithm 1.

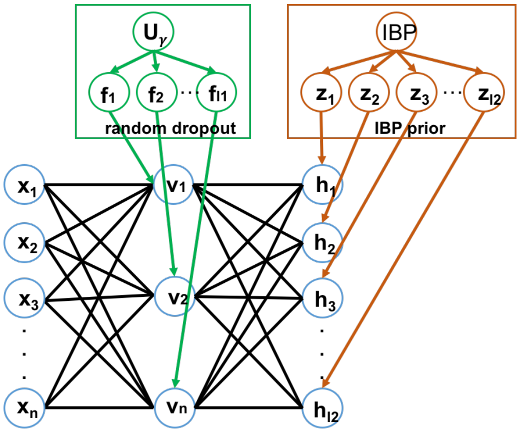

Figure 2: The structure of Heterogeneous Sparse DBN

4 EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we present our experiment setup and evaluate performance of our approach.

Experimental Procedure

We use the Exercise Activity dataset for our evaluation. The dataset contains human activities in different contexts and

Algorithm 1: Heterogeneous sparse DBN (HSparseDBN) training procedure

```
Input: Labeled dataset $D_{labeled} = \{x_i, y_i\}$, dropout rate
    $\gamma$, initial sample of $\pi$, learning rate $\lambda$
Output: Two layers deep belief network
    • Training procedure at first layer
    (1) Sample $h$ using Eq(2)
    (2) Remove a part of $h$ according to $f$, and sample $x$
        based on $h$ using Eq( 3)
    • Training procedure at second layer
    (1) Sample $\pi$ using Eq(12)
    (2) Sample $h^0, z^0|\pi, v^0$ using Eq(11)
    (3) Sample $v^1|\pi, h^0, z^0$ using Eq(10) and Eq(12)
    (4) Sample $h^1, z^1|\pi, v^1$ using Eq(11)
    • Back propagation on DBN
    (1) Update dropout and IBP layer parameters $\theta_{dropout}$
        and $\theta_{ibp}$ using Eq( 5)
```

have been recorded using tri-axial accelerometers. The sensor data is segmented using a sliding window with a size of 64 continuous samples and 50% overlap. We experiment with all our deep learning algorithms on a computer equipped with a Tesla K20c GPU and 64G memory. Other computations run on the same computer but on an Intel Xeon E5 CPU. Throughout this section, we use two hidden layer DBN. The dropout rate in the first hidden layer is 0.3 and the parameter values for IBP in the second layer are $\alpha = 1, \beta = 5$. The other parameters W, b, c in the network were initialized by drawing from a zero mean Gaussian with standard deviation 0.005. We also use weight decay and momentum in our networks. The regularization parameters are 0.998 and 0.95. We use rectified linear unit (ReLU) as the activation function.

Exercise Activity Dataset. In the Exercise Activity dataset [8], 20 test subjects were asked to perform a set of 10 exercise activities [6]. Each subject was equipped with three sensor-enabled devices: a Nexus S 4G phone in an armband, a MotoACTV wristwatch, and a second MotoACTV clipped to the hip. The dataset contains accelerometer and gyroscope data collected at 30 Hz sampling rate. For feature extraction, the sliding window size is empirically set to 1 second with 50% overlap based on a leave-one-out cross-validation test. The dataset contains around 8,000 instances.

Recognition Accuracy

In Table 1, we compare the accuracy values obtained for Exercise Activity dataset when using features learned from the training set using dropout DBN, Heterogeneous Sparse DBN (HSparseDBN), statistical features, and combinations of DBN with statistical features. The statistical features are obtained by calculating mean across the input using a sliding window. The classifier used here is a multi-class linearSVM [10]. When performing leave-one-out validation, only one user is used as test data and the rest form training data.

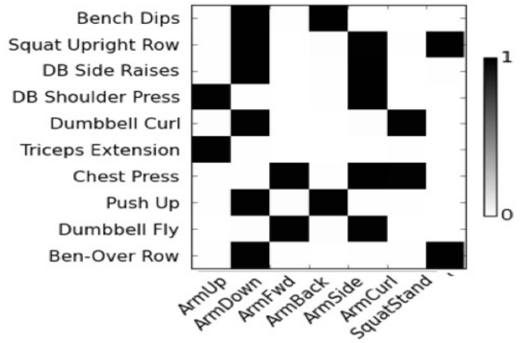

Figure 3: Correlation between human activities (in rows) and attributes (in columns) for the Exercise Activity dataset

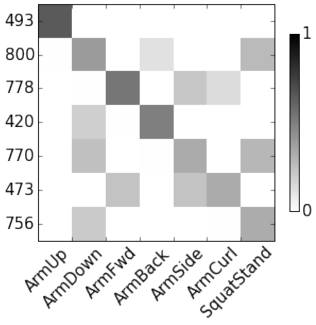

Figure 4: Correlation between DropoutDBN features and attributes for User 8

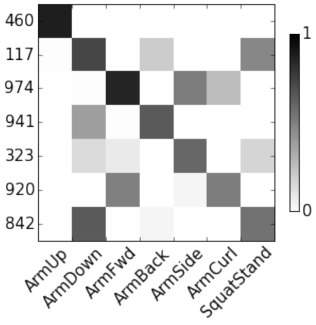

Figure 5: Correlation between HsparseDBN features and attributes for user 8

From Table 1, we see that accuracy is higher when using DBN features compared to using just statistical features. This suggests that DBN is able to capture more useful and relevant

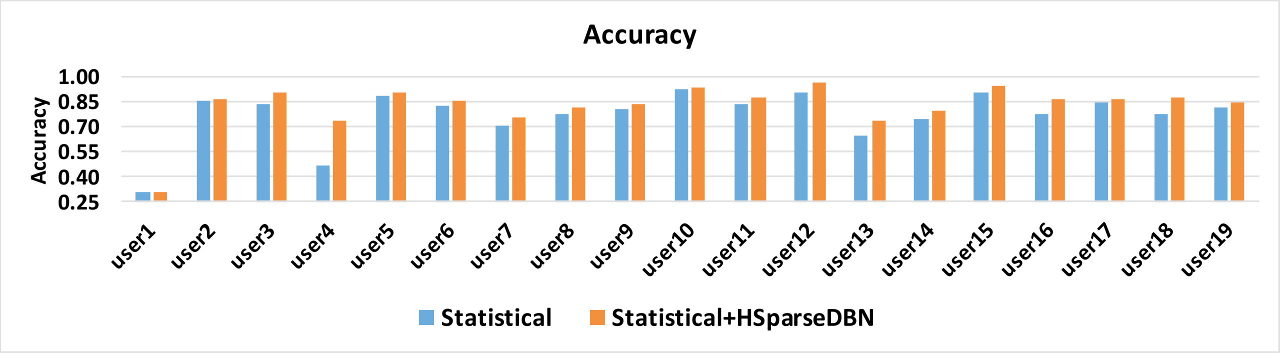

Figure 6: Accuracy of statistical features and statistical + HSparseDBN features, for users in Exercise Activity dataset

Table 1: Accuracy comparison between several methods, all using linearSVM classifier for the Exercise Activity dataset

Features	Accuracy
Statistical	77.30%
DropoutDBN	81.56%
HSparseDBN	82.30%
Statistical+DropoutDBN	84.38%
Statistical+HSparseDBN	**85.72%**

mid-level features. Furthermore, the accuracy of DBN + statistical features is higher than when using only DBN features or statistical features. This implies that DBN alone does not capture all of the features and that the statistical features are complementary to the DBN features. We also note that using HSparseDBN improves accuracy over DropoutDBN, thus demonstrating superior generalization ability of sparse features due to IBP's grouping characteristic. We conclude that the HSparseDBN + statistical features method benefits from both heterogenous sparsity of DBN and statistical features and hence outperforms all other methods.

Mid-Level Features Versus Semantic Attributes

In this section, we evaluate the degree to which the features learned using the DBN can capture semantic concepts. For each feature and label attribute pair, we compute the correlation and find the most correlated DBN-based feature for each semantic attribute.

$$r_{attri,dbn} = \frac{\sum_{i=1}^{M}(x_{attri} - x_{a\bar{t}tri})(x_{dbn} - \bar{x}_{dbn})}{\sqrt{\sum_{i=1}^{M}(x_{attri} - x_{a\bar{t}tri})^2 \sum_{i=1}^{M}(x_{dbn} - \bar{x}_{dbn})^2}} \tag{13}$$

Figure 4 show the correlation score of User 8. The features are represented by node numbers on the left. Most of dropout DBN features have a score greater than 0.5. The correlation between HSparseDBN features and attributes is shown in Figure 5. The result shows that all the corresponding features have high correlation scores. This supports our hypothesis that mid-level features from DBN can capture important relevant semantic concepts, and demonstrates the benefit of using heterogeneous sparsity.

Domain Adaption

An important aspect of evaluating the features is the degree to which they generalize well across different users, even if their distributions are different from each other. In this case the distribution of training set and test set is no longer i.i.d. In this subsection, we look into the accuracy of test user in the leave-one-out validation. In the test procedure, a test set contains certain instances from only one user, and the rest of users combine to form the training set. Since we already observed that Statistical+HSparseDBN performs best on average on these datasets, we compare the accuracy when using statistical features and the combination of statistical and HSparseDBN features. The results over 19 individual users are shown in Figure 6. From the results, we can see that statistical+HSparseDBN consistently outperforms statistical features alone, except for user 1 (31.07% vs. 30.72%).

5 CONCLUSION AND FUTURE WORK

In this paper, we demonstrate that deep neural networks can capture semantic concepts. We introduce a new approach for learning mid-level feature representation using dropout and Indian buffet process DBN, which can avoid overfitting and group similar features. We use Exercise Activity dataset for our experiments and are able to achieve promising results. We also study the semantic concepts by calculating the correlation between manually defined attributes and learned features, using which we show that many of the extracted mid-level features have semantic meanings. As future work, we will test the proposed method on more datasets and examine the inference process of semantic topics. We also intend on exploring the efficacy of Recurrent Neural Networks and their sequence modeling capability for learning mid-level features with semantic meanings.

REFERENCES

[1] Oresti Banos, Rafael Garcia, Juan A Holgado-Terriza, Miguel Damas, Hector Pomares, Ignacio Rojas, Alejandro Saez, and Claudia Villalonga. 2014. mHealthDroid: A Novel Framework for Agile Development of Mobile Health Applications. In *Proc. 6th International Work-Conference on Ambient Assisted Living*. Springer, 91–98.

[2] Ling Bao and Stephen S. Intille. 2004. Activity recognition from user-annotated acceleration data. In *Proc. International Conference on Pervasive Computing*. Springer, 1–17.

[3] Sourav Bhattacharya, Petteri Nurmi, Nils Hammerla, and Thomas Plötz. 2014. Using unlabeled data in a sparse-coding framework for human activity recognition. *Pervasive and Mobile Computing* 15 (2014), 242–262.

[4] Ulf Blanke and Bernt Schiele. 2010. Remember and transfer what you have learned-recognizing composite activities based on activity spotting. In *Proc. International Symposium on Wearable Computers*. 1–8.

[5] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research* 3 (2003), 993–1022.

[6] Keng-Hao Chang, Mike Y Chen, and John Canny. 2007. Tracking free-weight exercises. In *Proc. International Conference on Ubiquitous Computing*. Springer, 19–37.

[7] 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 *Proc. International Joint Conference on Pervasive and Ubiquitous Computing*. 355–358.

[8] 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 *Proc. International Conference on Mobile Systems, Applications, and Services*. 361–374.

[9] Snehal Chennuru, Peng-Wen Chen, Jiang Zhu, and Joy Ying Zhang. 2012. Mobile Lifelogger–Recording, Indexing, and Understanding a Mobile UserâĂŹs Life. In *Proc. International Conference on Mobile Computing, Applications, and Services*. 263–281.

[10] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research* 9 (2008), 1871–1874.

[11] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. 2009. Describing objects by their attributes. In *Proc. Conference on Computer Vision and Pattern Recognition*. 1778–1785.

[12] Zoubin Ghahramani, Thomas L Griffiths, and Peter Sollich. 2007. Bayesian nonparametric latent feature models. In *Proc. 8th World Meeting on Bayesian Statistics*.

[13] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier networks. In *Proc. International Conference on Artificial Intelligence and Statistics*, Vol. 15. 315–323.

[14] Geoffrey Hinton. 2002. Training products of experts by minimizing contrastive divergence. *Neural Computation* 14, 8 (2002), 1771–1800.

[15] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580* (2012).

[16] Tâm Huynh, Mario Fritz, and Bernt Schiele. 2008. Discovery of activity patterns using topic models. In *Proc. 10th International Conference on Ubiquitous Computing*. 10–19.

[17] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. 2009. Learning to detect unseen object classes by between-class attribute transfer. In *Proc. Conference on Computer Vision and Pattern Recognition*. 951–958.

[18] Nicholas D Lane and Petko Georgiev. 2015. Can Deep Learning Revolutionize Mobile Sensing?. In *Proc. 16th International Workshop on Mobile Computing Systems and Applications*. 117–122.

[19] Honglak Lee, Chaitanya Ekanadham, and Andrew Y Ng. 2007. Sparse deep belief net model for visual area V2. In *Proc. 20th International Conference on Neural Information Processing Systems*. 873–880.

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

[21] Roni Mittelman, Honglak Lee, Benjamin Kuipers, and Silvio Savarese. 2013. Weakly supervised learning of mid-level features with Beta-Bernoulli process restricted Boltzmann machines. In *Proc. Conference on Computer Vision and Pattern Recognition*. 476–483.

[22] Mark Palatucci, Dean Pomerleau, Geoffrey E Hinton, and Tom M Mitchell. 2009. Zero-shot learning with semantic output codes. In *Proc. 22nd International Conference on Neural Information Processing Systems*. 1410–1418.

[23] S. Pan, T. Yu, M. Mirshekari, J. Fagert, A. Bonde, O. J. Mengshoel, H. Y. Noh, and P. Zhang. 2017. FootprintID: Indoor Pedestrian Identification Through Ambient Structural Vibration Sensing. *Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3, Article 89 (September 2017), 31 pages.

[24] Huan-Kai Peng, Pang Wu, Jiang Zhu, and Joy Ying Zhang. 2011. Helix: Unsupervised grammar induction for structured activity recognition. In *Proc. 11th International Conference on Data Mining*. 1194–1199.

[25] Thomas Plötz, Nils Y Hammerla, and Patrick Olivier. 2011. Feature learning for activity recognition in ubiquitous computing. In *Proc. 22nd International Joint Conference on Artificial Intelligence*. 1729–1734.

[26] Olga Russakovsky and Li Fei-Fei. [n. d.]. Attribute learning in large-scale datasets. In *Trends and Topics in Computer Vision: First International Workshop on Parts and Attributes*. 1–14.

[27] Ruslan Salakhutdinov, Joshua B Tenenbaum, and Antonio Torralba. 2013. Learning with hierarchical-deep models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 8 (2013), 1958–1971.

[28] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.

[29] Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T Campbell. 2014. Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proc. International Joint Conference on Pervasive and Ubiquitous Computing*. 3–14.

[30] Pang Wu, Jiang Zhu, and Joy Ying Zhang. 2013. Mobisens: A versatile mobile sensing platform for real-world applications. *Mobile Networks and Applications* 18, 1 (2013), 60–80.

[31] T. Yu, Y. Zhuang, O. J. Mengshoel, and O. Yagan. 2016. Hybridizing Personal and Impersonal Machine Learning Models for Activity Recognition on Mobile Devices. In *Proc. 8th International Conference on Mobile Computing, Applications and Services*. 117–126. https://doi.org/10.4108/eai.30-11-2016.2267108

[32] M. Zeng, H. Gao, T. Yu, O. J. Mengshoel, H. Langseth, I. Lane, and X. Liu. 2018. Understanding and Improving Recurrent Networks for Human Activity Recognition by Continuous Attention. In *Proc. ACM International Symposium on Wearable Computers*. 56–63. https://doi.org/10.1145/3267242.3267286

[33] Ming Zeng, Le T Nguyen, Bo Yu, Ole J Mengshoel, Jiang Zhu, Pang Wu, and Joy Zhang. 2014. Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors. In *Proc. 6th International Conference on Mobile Computing, Applications and Services*. 197–205.

[34] M. Zeng, X. Wang, L. T. Nguyen, P. Wu, O. J. Mengshoel, and J. Zhang. 2014. Adaptive activity recognition with dynamic heterogeneous sensor fusion. In *Proc. 6th International Conference on Mobile Computing, Applications and Services*. 189–196.

[35] M. Zeng, T. Yu, X. Wang, L. T. Nguyen, O. J. Mengshoel, and I. Lane. 2017. Semi-supervised convolutional neural networks for human activity recognition. In *Proc. IEEE International Conference on Big Data*. 522–529.