MobileSOFT: U: A Deep Learning Framework to Monitor Heart Rate During Intensive Physical Exercise

Vasu Jindal
University of Texas, Dallas, TX
vasu.jindal@utdallas.edu

***Abstract*— Wearable biosensors have become increasingly popular in healthcare due to their capabilities for low cost and long term biosignal monitoring. However, current determination of heart rate through wearable devices and mobile applications suffers from high corruption of signals during intensive physical exercise. In this paper, we present a novel technique for accurately determining heart rate during intensive motion by classifying PPG signals obtained from smartphones or wearable devices combined with motion data obtained from accelerometer sensors. Our approach utilizes the Internet of Things (IoT) cloud connectivity of smartphones for selection of PPG signals using deep learning classification models. The technique is validated using the TROIKA dataset and is accurately able to predict heart rate with a 10-fold cross validation error margin of 4.88%.**

I. Introduction

A. Motivation & Prior Work

In the emerging era of Internet of Things (IoT) wearable devices like smart bands and smart watches e.g. Fitbit, Samsung Gear [1], etc., have gained interest in monitoring individual health metrics. Recent developments in biometrics have focused on the usage of physiological signals in heart rate calculation, biometric authentication and blood pressure calculation. These applications make use of physiological signals like electrocardiograms (ECG) [2], electroencephalograms (EEG) and photoplethysmograms (PPG) [3], [4]. These physiological signals offer many advantages for personalization and identification of individuals due to their capability of providing insights into clinical conditions.

Additionally, physiological signals like PPG can be collected for prolonged periods without extra effort using a wrist or a finger-tip based sensor. Currently, there are two popular options available for individual heart rate monitoring i) using applications on mobile phones like Azumio or ii) using a wrist based sensing device like Fitbit, Apple Watch [2]. Applications like Azumio are capable to determine heart rate in resting and controlled conditions but are inefficient during intensive aerobics or other forms of physical exercises. Most commercial wearable devices use embedded pulse oximeter for heart rate calculation. Recent developments also included using smartphone camera with flash to acquire PPG signals [3].

However, one challenge of using PPG signals in heart rate calculation is that PPG signals are highly susceptive to power line interference, respiration induced noise, muscle noise (EMG), motion artifact etc. Numerous signal processing techniques have been published to remove motion artifacts, such as adaptive noise cancelation, Kalman filter and independent component analysis [1]. Nevertheless, these traditional methods are inefficient in heavy physical exercise due to the overlap between frequency band of PPG and motion artifacts.

According to [5], factors like age, weight, gender as well as eating habits affect clinical metrics like heart rate, blood pressure etc. For example, a person who weighs over 80 kg may have a sharp increase in heart rate in a short interval of physical exercise while there is a much more gradual increase in a person weighing between 60 kg and 80 kg. This change in heart rate necessitates to create an adaptive model for identification of individuals into different groups and subsequently build personalized models to calculate individual health metrics.

In this paper, we propose a technique for heart rate calculation using smartphone accelerometer and PPG signal. We use the cloud connectivity of mobile phones to perform real time selection of PPG signals through deep belief networks and use these classified PPG signals for estimating heart rate. Mobile accelerometer is used to estimate the fundamental frequency and the noise produced by motion is removed through a noise canceler. Our technique is a three-step process, removing noise using power spectrum followed by a cloud based deep learning model to classify PPG using quality assessment. Fially, we we use a particle filter heart rate estimator to accurately show the heart rate during motion.

B. Prior Work

Heart-rate monitoring using PPG signals have been widely investigated in the literature. However, all prior works in PPG source identification are limited to PPG signals obtained from controlled (clinical) settings and highly accurate sensors. These restrictions may make these models unsuitable to use in day to day environments due to motion artifacts. Another technique presented in [6] investigated the feasibility of using PPG signals for heart-rate calculation. According to [6], researchers pre-processed the signal and extracted features using Linear Discriminant Analysis (LDA). The research concluded that signals should be collected in controlled environment with high precision sensors for source identification. This is a major shortcoming in existing approaches as in practical situations, subjects can be in various environments with different forms of motion. During the early summer of 2016, many lawsuits (http://www.wired.co.uk/article/fitbit-heart-rate-accuracy) were filed against Fitbit questioning the accuracy of the heart rate during intensive physical exercises

[10]. The study found that the devices were inaccurate by 20 beats per minute on average during moderate- to high-intensity exercise. The lawsuit concludes that the "devices could not provide meaningful heart rate data" when someone is exercising. While it is efficient in resting positions, devices like Fitbit and Apple watch are "advertised to people who exercise," and therefore, inaccurate heat rate measurements by such devices poses a high risk for individuals relying on these devices.

C. Main Contributions

In this paper we propose a two-stage technique to first cluster individuals into different groups. When a new individual uses the framework, his metrics and features are mapped to his/her identified group (cluster). Then, we present a signal processing and signal classification technique using Restricted Boltzman Machines and Deep Belief Networks. Additionally, we present several new discriminative features of the PPG signal. Our key contribution is to personalize training procedure for classification of human's PPG signal and use this data for customized calculation of clinical outcomes. We choose deep learning as the core basis of our approach due to its various advantages including fast interference and fast unsupervised learning. Deep learning has been intensively used in image recognition and voice source identification. To the best of our knowledge, it has never been applied to PPG signal classification.

II. Methodology

A. Dataset

The TROIKA framework presented in [7] is used to verify the robustness of our novel approach. The TROIKA dataset consists of an ECG signal, two-channel PPG signals and three-axis acceleration signals from 12 male subjects with ages ranging from 18 to 35. For each subject, the PPG signals were recorded from wrist by two pulse oximeters with green LEDs of wavelength 609 nm. To make the data recordings similar to practical world readings, the pulse oximeter and the accelerometer were embedded in a wristband and all signals were sampled at 125 Hz. During data recording, subjects walked or ran on a treadmill with the following speeds, in sequential order: 2 km/hour for 0.5 minute, 8 km/hour for 1 minute, 15 km/hour for 1 minute, 8 km/hour for 1 minutes, 15 km/hour for 1 minute, and 2 km/hour for 0.5 minute. Accurate heart-rate monoting using PPG should be possible even when the source is in heavy physical motion and henceforth, it is required to test learning models in practical day to day situations. Therefore, presence of different activities make TROIKA a suitable dataset for our model.

B. Overview of the Approach

Figure 1 illustrates an overview of our approach. The first steps includes noise cancellation using signal processing techniques. The device (Fitbit, Apple Watch) sends the PPG signal obtained to cloud for PPG pulse selection using deep belief networks in every 2 second time frame. Finally, if the PPG signal quality is classified within a reasonable threshold by the trained DBN then the observation is used to calculate heart rate. The heart rate estimation over time is achieved by applying particle filter on the classified PPG signals. In the end, calculated heart rate is stored in cloud and tagged with the associated exercise to increase accuracy of future predictions.

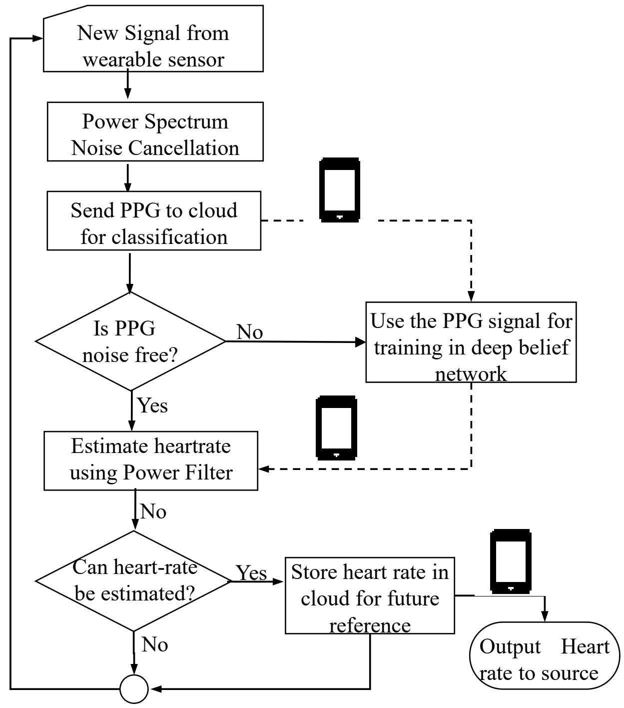

Fig. 1. Overview of Our Approach

C. Stage I.A: Preprocessing

PPG signals obtained from embedded pulse oximeters in commercial products are quite noisy due to motion artifacts. This noise needs to be removed for accurate identification. Furthermore, distinctive feature should be extracted to input in deep learning models. In our approach, the new PPG signal is obtained using a wearable sensor device like Fitbit or Apple Watch. The first step in the technique is noise cancellation and synthetic noise generation using power spectrum [4]. The accelerometer sensor of the mobile is used to obtain power spectrum density for all channels of signals. Normalization is performed on these spectrum to reduce error ratio. Subsequently, the power of the fused PPG at each possible hear rate frequency is determined by the maximal power of two-channel PPG signals at the same frequency. We then perform pre-processing stage including the following steps:

1. **Filtration**: The normalized PPG denoted by $PPG_{norm}(t)$ is calculated as following:

$$PPG_{norm}(t) = \frac{PPG_{raw}(t) - \mu_{PPG}(t)}{\sigma_{PPG}(t)} \tag{1}$$

where $\mu_{PPG}(t)$ and $\sigma_{PPG}(t)$ represents the mean and standard deviation of the PPG signal respectively. This normalized PPG signal is further processed with a moving average filter of length $F_c L$, where F_c is the frequency of the signal and L is a variable user parameter. Similar to [8], $L = 3$ was chosen to obtain the low-frequency trend. This low-trend is further removed using a 7th order Butterworth filter.

2. **Segmentation**: Segmentation of the PPG signal is necessary to extract discriminative features to input in classification models. We create segments each containing exactly one positive peak by using peak detection algorithms. The standard approach to find peaks by first deriving time series and identifying downward-going zero-crossing at peak maximums is incapable due to the random noise present in PPG. Therefore, we only consider zero crossings in the first derivative on two conditions: i) slopes exceeding the slope-threshold, ii) slope at locations where original signal exceeds the amplitude threshold. Essentially, each segment is a period of the signal.

3. **Interpolation and Extrapolation**: Interpolation and extrapolation are critical as segments obtained after the segmentation step may be of different lengths leading to inaccurate calculations of features in subsequent steps. Therefore, the segments are interpolated and extrapolated to a common length of $l = 125$ using Box-Jenkins AutoRegressive Moving Average (ARIMA) [9].

4. **Feature Extraction**: We used the following 11 features to be used as input for clustering and classification, a) Mean of PPG values in the segment b) Standard deviation of the values in the segment c) Average of the Dynamic Time Warp distance F_{dtw} between the chosen segment and all other segments, i.e.

$$F_{dtw} = \frac{\sum_{i=1}^{n} dtw(s_c, s_i)}{n}, i \neq c \tag{2}$$

where s_c represents the current chosen segment from n total segments and s_i for represents every other segment except the chosen segment, d) Maximum value of the segment, e) Minimum value of the segment, f) Position of maximum value within the segment, g) Position of minimum value within the segment, h) Maximum and minimum of computed discrete wavelet coefficients of the segment [10], j) Fisher-Pearson coefficient of skewness of the segment which we formulated as

$$g_1 = \frac{\sum_{i=1}^{n} (T_i - \overline{T})^3 / n}{s^3} \tag{3}$$

where $T_1, T_2, ..., T_n$ represents the PPG values in the segment, $\overline{T}$ is the mean, s is the standard deviation, and n is the number of data points (in our case 125), k) kurtosis of the PPG segment.

D. Stage I.B: Clustering Using PAM

In Stage I.B, individuals are separated into subgroups (clusters) before the final identification stage. We used Partitioning around Medoids (PAM) clustering method to separate sources into different groups by clustering the features of the source obtained from pre-processing stage. If a new source (not belonging to the dataset) is added to the set, we find its nearest neighbor and use the PPG heart-rate calculate model for its corresponding group. Separate groups based on different forms of motions as well as different types of sources (male, female, heavy-weight, under-weight) can be separated through this stage.

Given an initial set of sources it is often difficult to find the optimal number of clusters to create subgroups between the individuals. We address this shortcoming using GAP statistics [11] to estimate the initial number of clusters.

E. Stage II: Deep Learning to Classify PPG

The core of our technique uses the state-of-the-art deep learning for identification of individuals. Neural Networks with more than two layers allow the network to capture more complex patterns. We train deep learning classification models separately for each cluster as the groups may have been separated based on different forms of motion and different clinical metrics of individuals as described in the clustering step. Nevertheless, a common challenge in deep learning is overfitting which occurs when few training samples are used to train large models. We address this shortcoming, using a grid search approach to estimate the optimal size and number of hidden layers [12]. Our deep network contains one input layer, three hidden layers and one output layer for C_1 with hidden layers consisting of 40, 40 and 10 nodes respectively as shown in Figure 2. One input layer, three hidden layers and one output layer is utilized in models for both C_2 and C_3 with the hidden layers consisting of 30, 30 and 20 nodes.

The input of these classification models are the 11 feature vectors obtained after pre-processing stage. The activation function of hidden units is the sigmoid function which is traditionally used in nonlinear neural networks:

$$g(z) = \frac{1}{1 + exp(-z)} \tag{4}$$

In the function $z = \theta x$, where x is input and θ is the weight parameter. We further use a softmax classifier for output layer. This implies that the units in the output layer give the probability of the input features corresponding to the source. The output $h(x)$ of the softmax classifier for an input vector x is given by:

$$h_\theta(x) = \frac{1}{\sum_{j=1}^{K} \exp(\theta_j^\top x)} \begin{bmatrix} \exp(\theta_1)^\top x) \\ \exp(\theta_2)^\top x) \\ \vdots \\ \exp(\theta_k)^\top x) \end{bmatrix} \tag{5}$$

where K equals the number of final classes and θ is the weight of the parameter.

Another challenge in this technique is the higher number of parameters generally makes parameter estimation more difficult in neural networks. Therefore, it becomes extremely difficult to start the training of deep neural networks from random initial weight and bias values. We address this need and make our training faster and more feasible for day to day environments using a greedy unsupervised algorithm

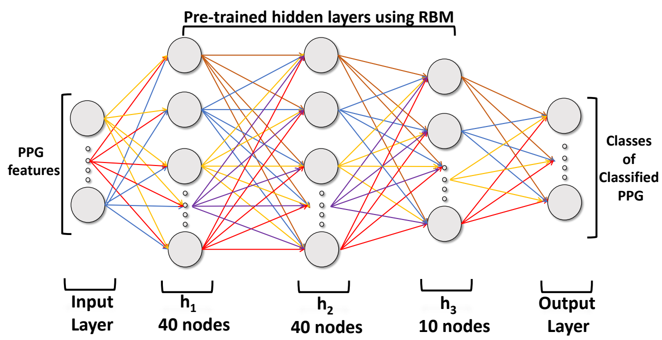

Fig. 2. Final Deep Belief Network for cluster C_1

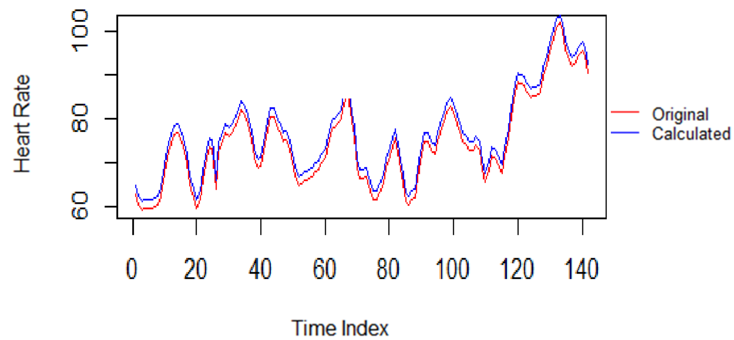

Fig. 4. Graph between the actual heart rate and predicted heart rate during low physical exercise (Average Error 3.84%)

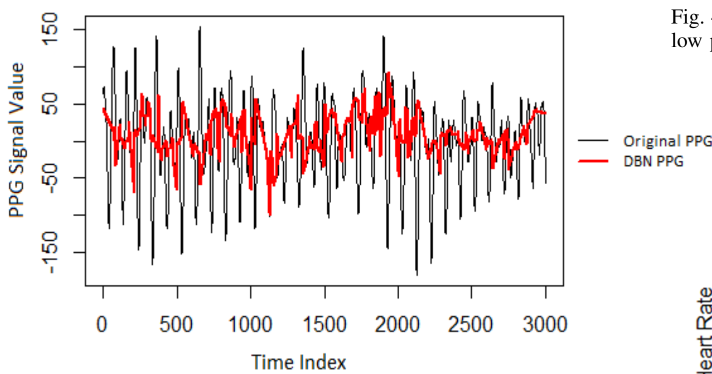

Fig. 3. Reconstructed PPG signal after pulse selection using Deep Belief Network

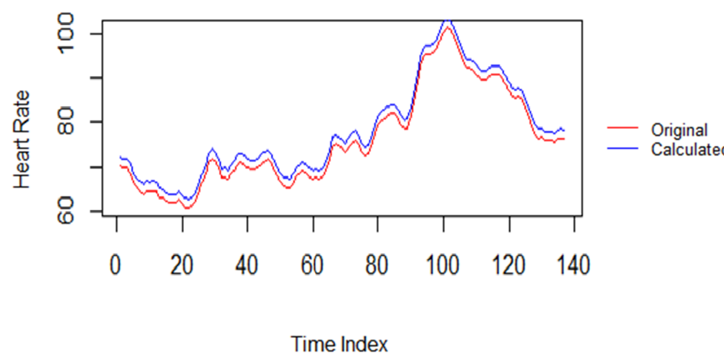

Fig. 5. Graph between the actual heart rate and predicted heart rate during intensive physical exercise (Average Error 5.72%)

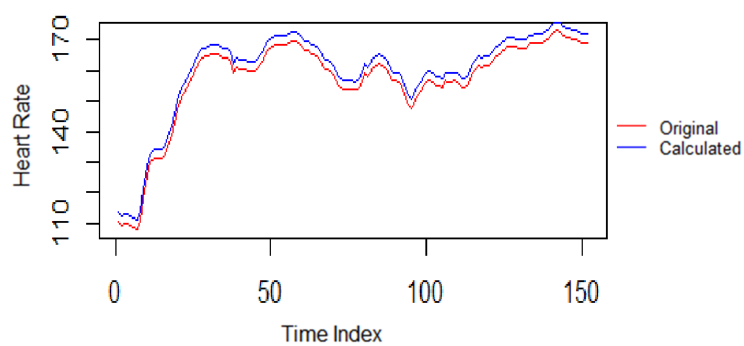

Fig. 6. Graph between the actual heart rate and predicted heart rate during boxing (Average Error 6.12%)

[13]. We further, incorporate Restricted Boltzmann Machines (RBM) to pre-train the network and find good initial weights for training deep belief networks. Briefly, RBMs are stochastic neural networks capable of predicting higher order and complex nonlinear input variables. Deep Belief Networks (DBN) are a class of Deep Neural Networks (DNN) with unsupervised pre-trained stacked Restricted Boltzmann Machines [14].

Let D_i be a DBN model for cluster C_i. In Stage II.A, the hidden layers of each D_i are first trained as RBMs using unlabeled inputs. We use Contrastive Divergence-1 (CD-1) [15] algorithm to obtain samples after 1-step Gibbs sampling. CD-1 allows accurate estimation of gradient's direction and minimize reconstruction error. This pre-training using unlabeled segmentations can be viewed as unsupervised learning and presents several advantages. First, due to this pre-training, D_i learns an identity function with same desired output as the original input. Secondly, it enhances the robustness of D_i by learning feature representations before the final supervised learning stage.

In Stage II.B, the pre-trained DBN network is fine-tuned by standard backpropagation with labeled segments as the input layer. Fine tuning is a critical step as it treats all pre-trained RBM layers as a single complete model and improves all weights in stacked RBMs in a single iteration. Figure 2

TABLE I
COMPARING OUR ALGORITHM AGAINST OTHER METHODS

Method	Approach	Setting of Collection	Personalization	Accuracy
[3]	Fuzzy Logic	Un-Controlled	No	82.3%
[6]	kNN	Controlled	No	74.0%
Ours	Clustering, Deep Learning	**Un-Controlled**	**Yes**	**96.1%**

represents the final resulting D_1 after tuning.

We perform PPG pulse selection on every 2 second time frame using our deep belief network (DBN). Figure 3 shows the reconstructed PPG from our deep learning model during a sample of 25 second period of intensive exercise.

III. Experimental Results

TROIKA [7] framework described earlier was used for all experiments to verify robustness of our model. e perform 10-fold cross validation to determine the accuracy of the predicted heart rate. Our technique determines heart rate during intensive exercise with an error rate of 4.88% Beats per Minute (BPM) calculated using ECG signal as the ground truth. Figure 4,5 and 6 show a sources predicted heart rate using our technique with actual heart rate obtained from ECG sensor positioned near the heart in the TROIKA dataset. The model learns the clinical metrics associated with the source and stores results in the cloud for future analysis. Table I shows the comparison of our approach with previously published approaches. Unlike previous techniques which are inefficient in intensive physical activities, our approach specifically addresses this issue through training models personalized both the activity performed and each persons clinical metrics for final heart rate calculation.

IV. Conclusions

This paper presents a novel approach in Internet of Things (IoT) to calculate heart rate using smartphone accelerometer and camera as well as wearable devices through personalized deep learning models. Currently, our deep belief network only use features from motion data obtained from accelerometer and recorded PPG signals. Future work includes the extension of this approach for PPG signal selection by extracting new features from remote PPG [2].

References

[1] J. Takacs, C. L. Pollock, J. R. Guenther, M. Bahar, C. Napier, and M. A. Hunt, "Validation of the fitbit one activity monitor device during treadmill walking," *Journal of Science and Medicine in Sport*, vol. 17, no. 5, pp. 496–500, 2014.

[2] A. Lourenço, H. Silva, and A. Fred, "Unveiling the biometric potential of finger-based ecg signals," *Computational intelligence and neuroscience*, vol. 2011, p. 5, 2011.

[3] Y. Gu, Y. Zhang, and Y. Zhang, "A novel biometric approach in human verification by photoplethysmographic signals," in *4th International IEEE EMBS Special Topic Conference on Information Technology Applications in Biomedicine*, 2003, pp. 13–14.

[4] Y. Gu and Y. Zhang, "Photoplethysmographic authentication through fuzzy logic," in *IEEE EMBS Asian-Pacific Conference on Biomedical Engineering*, 2003, pp. 136–137.

[5] J. Hirsch, R. L. Leibel, R. Mackintosh, and A. Aguirre, "Heart rate variability as a measure of autonomic function during weight change in humans," *American Journal of Physiology-Regulatory, Integrative and Comparative Physiology*, vol. 261, no. 6, pp. R1418–R1423, 1991.

[6] P. Spachos, J. Gao, and D. Hatzinakos, "Feasibility study of photoplethysmographic signals for biometric identification," in *17th IEEE International Conference on Digital Signal Processing (DSP)*, 2011, pp. 1–5.

[7] Z. Zhang, Z. Pi, and B. Liu, "Troika: A general framework for heart rate monitoring using wrist-type photoplethysmographic signals during intensive physical exercise," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 2, pp. 522–531, 2015.

[8] E. Grisan, G. Cantisani, G. Tarroni, S. K. Yoon, and M. Rossi, "A supervised learning approach for the robust detection of heart beat in plethysmographic data," in *37th IEEE Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015, pp. 5825–5828.

[9] G. E. Box and D. A. Pierce, "Distribution of residual autocorrelations in autoregressive-integrated moving average time series models," *Journal of the American statistical Association*, vol. 65, no. 332, pp. 1509–1526, 1970.

[10] D. Cvetkovic, E. D. Übeyli, and I. Cosic, "Wavelet transform feature extraction from human ppg, ecg, and eeg signal responses to elf pemf exposures: A pilot study," *Digital signal processing*, vol. 18, no. 5, pp. 861–874, 2008.

[11] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 2, pp. 411–423, 2001.

[12] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 281–305, 2012.

[13] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle *et al.*, "Greedy layer-wise training of deep networks," *Advances in Neural Information Processing Systems*, vol. 19, p. 153, 2007.

[14] N. Le Roux and Y. Bengio, "Representational power of restricted boltzmann machines and deep belief networks," *Neural computation*, vol. 20, no. 6, pp. 1631–1649, 2008.

[15] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural computation*, vol. 14, no. 8, pp. 1771–1800, 2002.