

TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

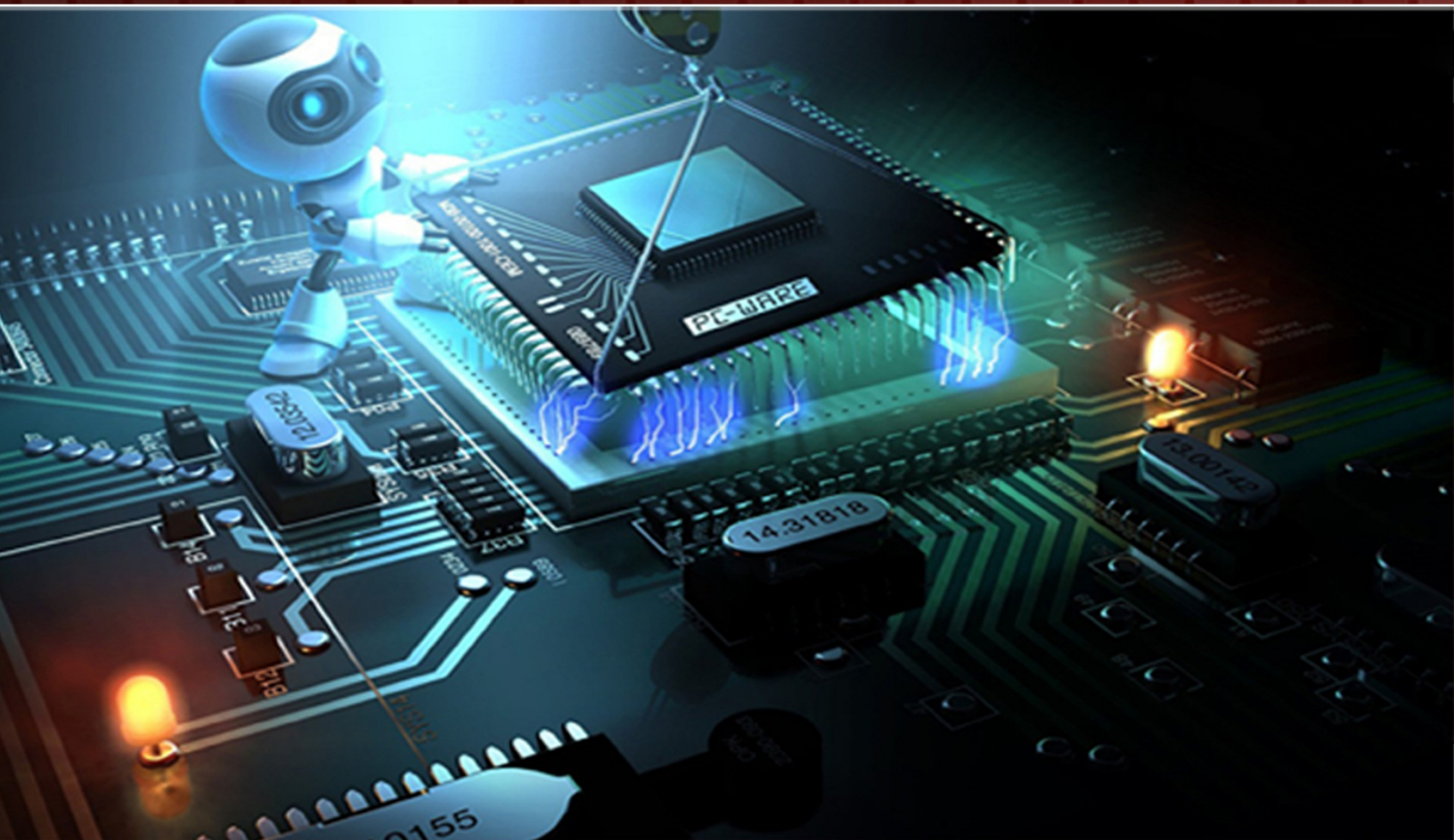


TABLE OF CONTENTS

EDITORIAL ADVISORY BOARD	I
DISCLAIMER	II
Enhancing Single Speaker Recognition Using Deep Belief Network Murman Dwi Prasetyo, Tomohiro Hayashida, Ichiro Nishizaki, Shinya Sekizaki	1
Artificial Human Optimization – An Overview Satish Gajawada	21
Development of an Electronic Nose for Olfactory System Modelling using Artificial Neural Network Mary Anne Roa, Proceso Fernandez	30

EDITORIAL ADVISORY BOARD

Editor-in-Chief

Professor Er Meng Joo
Nanyang Technological University
Singapore

Members

Professor Djamel Bouchaffra
Grambling State University, Louisiana
United States

Prof Bhavani Thuraisingham
The University of Texas at Dallas
United States

Professor Dong-Hee Shin,
Sungkyunkwan University, Seoul
Republic of Korea

Professor Filippo Neri,
Faculty of Information & Communication Technology
University of Malta
Malta

Prof Mohamed A Zohdy,
Department of Electrical and Computer Engineering
Oakland University
United States

Dr Kyriakos G Vamvoudakis,
Dept of Electrical and Computer Engineering
University of California Santa Barbara
United States

Dr Luis Rodolfo Garcia
College of Science and Engineering
Texas A&M University, Corpus Christi
United States

Dr Hafiz M. R. Khan
Department of Biostatistics
Florida International University
United States

Dr. Xuewen Lu
Dept. of Mathematics and Statistics
University of Calgary
Canada

Dr. Marc Kachelriess
X-Ray Imaging and Computed Tomography
German Cancer Research Center
Germany

Dr. Nadia Pisanti
Department of Computer Science
University of Pisa
Italy

Dr. Frederik J. Beekman
Radiation Science & Technology
Delft University of Technology, *Netherlands*

Professor Wee SER
Nanyang Technological University
Singapore

Dr Xiacong Fan
The Pennsylvania State University
United States

Dr Julia Johnson
Dept. of Mathematics & Computer Science
Laurentian University, Ontario
Canada

Dr Chen Yanover
Machine Learning for Healthcare and Life Sciences
IBM Haifa Research Lab
Israel

Dr Vandana Janeja
University of Maryland, Baltimore
United States

Dr Nikolaos Georgantas
Senior Research Scientist at INRIA, Paris-Rocquencourt
France

Dr Zeyad Al-Zhour
College of Engineering, The University of Dammam
Saudi Arabia

Dr Zdenek Zdrahal
Knowledge Media Institute,
The Open University, Milton Keynes
United Kingdom

Dr Farouk Yalaoui
Institut Charles Dalaunay
University of Technology of Troyes
France

Dr Jai N Singh
Barry University, Miami Shores, Florida
United States

Dr. Laurence Devillers
Computer Science, Paris-Sorbonne University
France

Dr. Hans-Theo Meinholz
Systems analysis and middleware
Fulda University of Applied Sciences
Germany

Dr. Katsuhiko Honda
Department of Computer science and Intelligent Systems
Osaka Prefecture University
Japan

Dr. Uzay Kaymak
Department of Industrial Engineering & Innovation
Sciences, Technische Universiteit Eindhoven University of
Technology, *Netherlands*

-
- Dr. Ian Mitchell**
Department of Computer Science
Middlesex University London
United Kingdom
- Dr. Weiru Liu**
Department of Computer Science
University of Bristol
United Kingdom
- Dr. Aladdin Ayesh**
School of Computer Science and Informatics
De Montfort University, Leicester
United Kingdom
- Dr. David Glass**
School of Computing, Ulster University
United Kingdom
- Dr. Sushmita Mukherjee**
Department of Biochemistry
Weil Corner Medical College, New York
United States
- Dr. Rattikorn Hewett**
Dept. of Computer Science
Texas Tech University
United States
- Dr. Cathy Bodine**
Department of Bioengineering
University of Colorado
United States
- Dr. Daniel C. Moos**
Education Department
Gustavus Adolphus College
United States
- Dr. Anne Clough**
Department of Mathematics,
Statistics and Computer Science, Marquette University
United States
- Dr. Jay Rubinstein**
University of Washington
United States
- Dr. Frederic Maire**
Department of Electrical Engineering and Computer Science
Queensland University of Technology
Australia
- Dr. Bradley Alexander**
School of Computer Science
University of Adelaide
Australia
- Dr. Erich Peter Klement**
Department of Knowledge-Based Mathematical Systems
Johannes Kepler University Linz
Austria
- Dr. Ibrahim Ozkan**
Department of Economics
- Dr. Fernando Beltran**
University of Auckland Business School
New Zealand
- Dr. Mikhail Bilenko**
Machine Intelligence Research (MIR) Group, Yandex
Russia
- Anne Hakansson**
Department of Software and Computer systems
KTH Royal Institute of Technology
Sweden
- Dr. Adnan K. Shaout**
Department of Electrical and Computer Engineering
University of Michigan-Dearborn
United States
- Dr. Tomasz G. Smolinski**
Department of Computer and Information Sciences
Delaware State University
United States
- Dr. Yi Ming Zou**
Department of Mathematical Sciences
University of Wisconsin
United States
- Mohamed A. Zohdy**
Department of Electrical and Systems Engineering
Oakland University
United States
- Dr. Krysta M. Svore**
Microsoft Quantum – Redmond Microsoft
United States
- Dr. John Platt**
Machine learning, Google
United States
- Dr. Wen-tau Yih**
Natural language processing
Allen Institute for Artificial Intelligence
United States
- Dr. Matthew Richardson**
Natural Language Processing Group, Microsoft
United States
- Amer Dawoud**
Department of Computer Engineering
University of Southern Mississippi
United States
- Dr. Jinsuk Baek**
Department of Computer Science
Winston-Salem State University
United States
- Dr. Harry Wechsler**
Department of Computer Science
George Mason University
United States

Hacettepe University
Canada

Dr. Sattar B. Sadkhan
Department of Information Networks
University of Babylon
Iraq

Dr. Marina Papatriantafilou
Department of Computer Science and Engineering
Chalmers University of Technology
Sweden

Dr. Florin Manea
Dependable Systems Group, Dept. of Computer Science
Kiel University Christian-Albrechts
Germany

Prof. Dr. Hans Kellerer
Department of Statistics and Operations Research
University of Graz
Austria

Dr. Dimitris Fotakis
School of Electrical and Computer Engineering
National Technical University of Athens
Greece

Dr. Faisal N. Abu-Khzam
Department of Computer Science and Mathematics
Lebanese American University, Beirut
Lebanon

Dr. Tatsuya Akutsu
Bioinformatics Center, Institute for Chemical Research
Kyoto University, Gokasho
Japan

Dr. Francesco Bergadano
Dipartimento di Informatica,
Università degli Studi di Torino
Italy

Dr. Mauro Castelli
NOVA Information Management School (NOVA IMS),
Universidade Nova de Lisboa
Portugal

Dr. Stephan Chalup
School of Electrical Engineering and Computing,
The University of Newcastle
Australia

Dr. Louxin Zhang
Department of Mathematics, National University of
Singapore

Dr. Omer Weissbrod
Department of Computer Science
Israel Institute of Technology
Israel

Dr. Yael Dubinsky
Department of Computer Science
Israel Institute of Technology
Israel

Dr. Francesco Bergadano
Department of Computer Science
University of Turin
Italy

Dr. Marco Chiarandini
Department of Mathematics and Computer Science,
University of Southern
Denmark

Dr. Xiaowen Chu
Department of Computer Science, Hong Kong Baptist
University, Kowloon Tong
Hong Kong

Dr. Alicia Cordero
Instituto de Matemática Multidisciplinar, Universitat
Politècnica de València
Spain

Dr. Sergio Rajsbaum
Instituto de Matemáticas, Universidad Nacional
Autónoma de México
Mexico

Dr. Tadao Takaoka
College of Engineering
University of Canterbury, Christchurch
New Zealand

Dr. Bruce Watson
FASTAR Group, Information Science Department,
Stellenbosch University
South Africa

Dr. Tin-Chih Toly Chen
Department of Industrial Engineering and Management,
National Chiao Tung University, Hsinchu City
Taiwan

DISCLAIMER

All the contributions are published in good faith and intentions to promote and encourage research activities around the globe. The contributions are property of their respective authors/owners and the journal is not responsible for any content that hurts someone's views or feelings etc.

Enhancing Single Speaker Recognition Using Deep Belief Network

¹Murman Dwi Prasetyo, ¹Tomohiro Hayashida, ¹Ichiro Nishizaki, ¹Shinya Sekizaki

¹Graduate School of Engineering, Hiroshima University, Hiroshima, JAPAN;

prasetyo_arel.corp@yahoo.com; d161685@hiroshima-u.ac.jp

ABSTRACT

Recognition in speech is complex phenomena study, and the reason for this is the complexity of human language. The barrier of the problem in speech recognition study now can be handled from speech signal using machine learning methods. Nowadays, Deep Belief Networks (DBN) automatically is able to find out the representation of speech signal.

This paper tries to approach a structure optimization of DBN which based on the combined technique of evolutionary computation to enhance the single speaker speech. It firstly extracts from the feature of speech signal then applies them to construct lots of random subspaces. The result of the conducted experimental in the evolutionary computation of DBN indicates the structure have an improvement for speech recognition.

Keywords: Deep Belief Network; Evolutionary Computation; Speech Recognition; Speech Signal; Random Subspace.

1 Introduction

Speech is the audio form for communication in human behaviors, which is spoken continuously states. In the speech recognition the continuously spoken states convert an acoustic signal, it captured by a microphone or telephone to a length of words. The characteristics of signal it reflects the different speech sound being spoken. The information from a speech that we are gathering is represented by a spectrum of amplitude from speech waveform. Based on this speech characteristic allows us to recognize the feature information from the waveform of the speech signal. Recognizing word from the acoustic signal is a though work, many researchers are emerging in the area of speech recognition and signal processing [1].

Speech Recognition as well-known as computer speech recognition is the process learning from the computer to understand our spoken by an algorithm implemented as a computer program. The main goal of the speech recognition area is developing speech recognition technology and systems into machines. The basic communication from a human being is a speech of human speech ability of the machine, the desire to automate simple tasks requiring machine interaction with humans in an automatic speech [2].

Nowadays, the application in tasks that require human-machine interfaces, such as automatic call processing in telephone networks, and query-based information systems find widespread in the statistical modeling of speech, automatic speech recognition systems. That provides updated travel information, stock price quotations, weather reports, data entry, voice dictation, access to information: travel, banking,

commands, transcription, disabled people (blind people) supermarket, railway reservations, etc. [3]. Speech recognition technology was increasingly used within telephone networks to automate as well as to enhance the operator services [4].

In the sixth decades of speech, recognition area has attracted many researchers' attention for the reason of curiosity about technology and the mechanism of realization. The speech feature extraction is a key issue for all classification methods to obtain better generalization. The extracted features should minimize the distances between samples with the same speech class and maximize the distances between samples with the different speech classes [5]. If the features are not well defined, the best classifier could have difficulty in reaching the good performance. Most typical features are predefined by hand-engineered ones, including newly proposed nonlinear dynamic features [7].

They have achieved the great success in specific fields where the small speech training data can be available only. However, these features perform inconsistently on different speech recognition tasks [8]. They are on the lower level to make themselves difficult to extract and organize the discriminative features from the speech signals. It is not clear which speech features are most powerful in distinguishing recognition [2, 8]. They are easily influenced by speakers, speaking styles, sentences, and speaking rates because these factors directly affect the extracted speech features such as pitch and energy contours [5]. Besides, they are not easily tuned for the newly coming speech signals-pitch and energy contours [5]. Besides, they are not easily tuned for the newly coming speech signals.

Recently, the development of machine learning based on speech has been made in a deep neural network similar as a neural network. One of approaching algorithm in deep neural networks is Deep Belief Network (DBN) [3]. For example, speech signal utilizes the higher level features to represent the more abstract concepts [10]. This is the reason that they succeed in breaking most of the world records of the recognition tasks. Among deep learning methods, deep belief network (DBN) is the most representative one [11, 12]. It applies the unsupervised learning algorithms such as auto-encoders and sparse coding to learn higher level feature representations from the unlabeled data [13]. It has produced the state-of-the-art results on recognition and classification tasks [10]. On the other hand, typical classification methods used for speech recognition include hidden Markov model (HMM) [14], Gaussian Mixture Model (GMM) [15], artificial neural networks such as recurrent neural network (RNN) [16], support vector machine (SVM) [17, 18], and the fuzzy cognitive map network [19]. These methods are confronted with the complicated decision boundary of the classification.

In such case, the ensemble learning can be applied that can learn any nonlinear boundary through appropriately combining the simple classifiers. It has potential ability to reduce over fitting problems greatly, to decrease the risk of a single classifier, and to obtain better performance than its single classifiers [20]. The usual ensemble classifiers are boost-based, bagging-based approaches [21], random subspace [22], and so forth. Some of them have been applied to perform speech recognition but still fail to reach the performance as expected. For example, it seems that random forest and AdaboostDT have the bad effect for speech classification [23]. The possible reason is that the diversity of the base classifiers is not guaranteed [24]. As to random subspace, the classifiers trained with different features should have certain diversity inherently. However, in the neural networks (NN) are prone to over fitting. Especially, the deep neural networks in some cases where the training data are not abundantly clear [24]. For instance, there are two different features sets, but the classifiers trained by the two features sets may

have the similar classification results, leading to no rich diversity between them [24]. To ensure the diversity among base classifiers, the features in random subspace should be further abstracted from different viewpoints using DBN.

This paper presents an evolutionary computational method for speech recognition, which is composed of the DBN and Tabu Search. Hayashida et al. [25] describes a number of subspace the implementation of tabu search is applied. Each subspace can be directly fed into DBN to generate the high-level features. The rest of this paper is organized as follows: In Section 2, several related works are briefly introduced about speech recognition techniques, DBN and RBM. The evaluated system and some experiments on simple voice dataset are presented in Section 3. Then Section 4 describes about the results and discussion. Finally, Section 5 concludes this paper.

2 Speech Techniques and Structural Optimization

2.1 Speech Recognition.

This section will introduce about speech recognition technique.

Speech is a moving signal. When we speak, our articulatory apparatus (the lips, jaw, tongue, and velum) modulates the air pressure and flow to produce an audible sequence of sounds [12]. Although the spectral content of any particular acoustic signal in speech may include sequences up to several thousand hertz, our articulatory configuration (vocal-tract shape, tongue movement, etc.) often does not undergo dramatic changes more than 10 times per second [13]. The acoustic properties of a waveform corresponding to a phone can vary greatly depending on many factors - phone context, speaker, style of speech, etc.

The main process in the speech recognition is feature extraction, it would be reduced variability of spoken words signal. Particularly, eliminating various information, such as whether the sound is voiced or unvoiced, it eliminates the effect of periodicity or pitch, the amplitude of excitation signal and also the fundamental frequency etc. The feature extraction techniques for speech recognition describes about reducing dimensionality of input vector while maintaining the discriminating power of signal. Many researchers have some point of important work in speech recognition area [14]. The related works for speech recognition techniques follows by:

- Principle Component Analysis (PCA)
- Independent Component Analysis (ICA)
- Mel-Frequency Scale Analysis
- Filter Bank Analysis
- Mel-Frequency Cepstrum (MFCC)
- Integrated Phoneme Subspace method (PCA, LDA and ICA)

Recently, the common technique in order to desire processing task in speech recognition is MFCC. The characteristics such as peak, pitch spectrum mean and standard deviation of the signal are extorted from denoised signal [15].

All modern descriptions of speech are to some degree probabilistic. That means that there are no certain boundaries between units, or between words. Speech to text translation and other applications of speech are never 100% correct. That idea is rather unusual for software developers, who usually work with deterministic systems. And it creates a lot of issues specific only to speech technology [16]. Therefore, a

number of problems with the standard method of hidden Markov model (HMM) and features that come from fixed and frame based spectra (eg MFCC) are discussed [17].

On the other hand, the approaches method for speech recognition is acoustic phonetic, pattern recognition, artificial intelligence [26]. Acoustic approach that stated by hemdal and hughes 1967 [27], this method was based on finding speech sound and label it.

The process of acoustic approach is analyst the spectral from speech combined with feature detection to set of feature that describe the wide acoustic properties. Then, segmenting and labeling the speech signal becomes a stable acoustic area followed by one or more phonetic labels that produces the result in phoneme lattice characterization in speech. Finally, in this approach tries to determine the correct words or string of word.

Next approach is pattern recognition. This method is known generically as pattern matching. In they studied Itakura 1975; Rabiner 1989; Rabiner and Juang 1993) [28] pattern matching involves two essential step; this step was namely pattern training and pattern comparison. The most important feature of this approach is that it uses well-formulated mathematical structures and determines the representation of appropriate utterance patterns and a reliable pattern comparison of a set of labeled training samples through a formal training algorithm. The most widely used statistical method in pattern recognition is Hidden Markov Model (HMM).

In 1994, Moore [29] studies about template based approach. The underlying idea of this method is easy to understand. The prototype collection of speech patterns is stored as a reference pattern representing the dictionary of candidate words. At this stage the algorithm will match unknown speech utterances with each template reference and choose the best category as the matching pattern. Typically, templates for all words are built This has the advantage of it, because of less acoustic segmentation fault or classification of more variable units such as phonemes can be avoided. The other main idea is to use a dynamic form of programming approach to temporarily align the pattern to account for the difference in speech levels across the speaker as well as the repetition of words in the same way the speaker. For more detail about the speech recognition techniques, see table 1.1

Table 1. Speech Techniques

Approach	Representation	Recognize Function	Typical
Acoustic Phonetic Pattern Recognition:	Phonemes	Lexical Probability	Log Likelihood Ratio
	● Template Pixel from speech samples	Correlation distance	Classification error
	● DTW Spectral sequences	Dynamic wrapping algorithm	Euclidian Distance
	● Statistical Spectral vectors	Clustering Function	Classification error
Neural Network	Speech Features	Network function	Mean Square Error (MSE)
SVM	Kernel	Max Margin	Minimizing bound of error

2.2 Speech Recognition Development (year wise).

In 1950 the researchers tried to exploit the basic idea of phonetic acoustics in the earliest attempt to design a system that could communicate with machines. During the 1950s, most voice recognition systems learned about spectral resonance over the span of each utterance extracted from the signal output of the bank's analog filter and the logic circuit [30]. In 1952, at Bell's laboratory, Davis, et.al built a system for the introduction of isolated digits for one speaker [31]. In this system is very dependent on the measurement of resonance time on the pronunciation.

In an independent venture at RCA Laboratories in 1956, Olson and Belar tried to recognize 10 different syllables from one speaker, embodied in 10 two-syllable words [32]. The system also relies on spectral measurements (already available analog filter banks) especially during vulnerable conversations. In 1959, at University College in England, Fry and Denes tried to build phoneme recognition to recognize four vowels and nine consonants [34]. Actually the development of speech recognition is significantly fast. In this paper development information of speech recognition recognize start at 2009-2017s.

2.2.1 2009-2017s

In the early 2009, the development of speech recognition technology becomes inexpensive and powerful. Nowadays, the development of technology is supported by advances in artificial intelligence and the increasing number of data words that can be easily mined, it is possible the development of technology has recently become the next dominant interface.

In the long history of speech recognition, both shallow form and deep form (e.g. recurrent nets) of artificial neural networks had been explored for many years during 1980s, 1990s and a few years into the 2000s.[35][36][37] But these methods never won over the non-uniform internal handcrafting Gaussian mixture model/Hidden Markov model (GMM-HMM) technology based on generative models of speech trained discriminatively.[38] A number of key difficulties had been methodologically analyzed in the 1990s, including gradient diminishing[39] and weak temporal correlation structure in the neural predictive models.[40][41].

In contrast to HMMs, neural networks make no assumptions about feature statistical properties and have several qualities making them attractive recognition models for speech recognition. When used to estimate the probabilities of a speech feature segment, neural networks allow discriminative training in a natural and efficient manner. Few assumptions on the statistics of input features are made with neural networks. However, in spite of their effectiveness in classifying short-time units such as individual phonemes and isolated words, [41] neural networks are rarely successful for continuous recognition tasks, largely because of their lack of ability to model temporal dependencies.

All these difficulties were in addition to the lack of big training data and big computing power in these early days. Most speech recognition researchers who understood such barriers hence subsequently moved away from neural nets to pursue generative modeling approaches until the recent resurgence of deep learning starting around 2009–2010 that had overcome all these difficulties. Hinton et al. and Deng et al. reviewed part of this recent history about how their collaboration with each other and then with colleagues across four groups (University of Toronto, Microsoft, Google, and IBM) ignited a renaissance of applications of deep feed-forward neural networks to speech recognition. [42] [43] [44] [45].

Today [47], A Microsoft research executive called this innovation "the most dramatic change in accuracy since 1979." [48] In contrast to the steady incremental improvements of the past few decades, the application of deep learning decreased word error rate by 30%. [46] This innovation was quickly adopted across the field. Researchers have begun to use deep learning techniques for language modeling as well. A deep feed-forward neural network (DNN) is an artificial neural network with multiple hidden layers of units between the input and output layers. [44] Similar to shallow neural networks, DNNs can model complex nonlinear relationships. DNN architectures generate compositional models, where extra layers enable composition of features from lower layers, giving a huge learning capacity and thus the potential of modeling complex patterns of speech data [49].

A success of DNNs in large vocabulary speech recognition occurred in 2010 by industrial researchers, in collaboration with academic researchers, where large output layers of the DNN based on context dependent HMM states constructed by decision trees were adopted. [50][51] [52] See comprehensive reviews of this development and of the state of the art as of October 2014 in the recent Springer book from Microsoft Research. [53] See also the related background of automatic speech recognition and the impact of various machine learning paradigms including notably deep learning in recent overview articles [54][55].

One fundamental principle of deep learning is to do away with hand-crafted feature engineering and to use raw features. This principle was first explored successfully in the architecture of deep auto encoder on the "raw" spectrogram or linear filter-bank features, [56] showing its superiority over the Mel-Cepstral features which contain a few stages of fixed transformation from spectrograms. The true "raw" features of speech, waveforms, have more recently been shown to produce excellent larger-scale speech recognition results [57].

2.3 Neural Networks Involve into Deep Belief Network (DBN).

This section will introduce about several models of neural networks involvement.

Since a decade the development of artificial neural networks has been used in artificial technology applications. It has been consisted of pattern recognition, voice and speech analysis and natural language processing. Due to lack of effectiveness of networks performance in Neural Network some cases, deep models and architectures with many layers were suggested. The theorem of Deep Belief Network (DBN) is first introduced by Hinton et.al [8]. DBN is a basic network that consists of belief network composed of multiple layers of Restricted Boltzmann Machines (RBM) [9]. In DBN feature extraction works under unsupervised learning, it's called pre-training process then fine-tuning is performed in remaining process. Figure. 1 shows DBN consisting of three layers of RBM as the standard models.

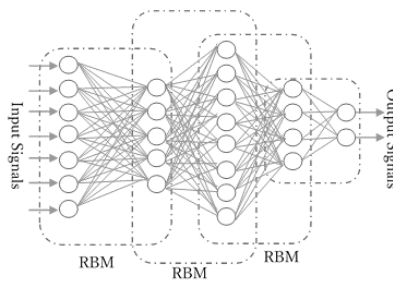


Figure 1. Standard Deep Belief Network Model

2.3.1 Deep Belief Network in RBM

An invention of Boltzmann Machine was first introduced by Smolensky in 1986 [21] with the characteristic of one hidden layer and one visible layer. In 2000, Hinton et.al improved the Boltzmann Machine into Restricted Boltzmann Machine which is used as generative models of different types of data. A Restricted Boltzmann Machine models consist of two sets of units visible and hidden units. A visible unit can be derived as the units that are directly observed in RBM. Hidden units are not directly connected to training data but the models dependencies between components. This structure of RBM can adjust the parameter in order to make the probability distributions of RBM fits in the training data as much as possible.

To understand bipartite graph of RBM, we have commonly example of study [8], Hinton et.al conducts the experiment of DBN in Modified National Institute of Standards and Technology (MNIST) database. In his experiment the binary images of MNIST as training set for RBM. The training of MNIST model are represented random binary pixels as visible units and the connected of stochastic binary feature vectors modeled as hidden units. In this paper work, we would like try to conduct similar experiment like MNIST but in the speech format database.

$$E(v, h) = - \sum_j b_j v_j - \sum_i c_i h_i - \sum_j \sum_i v_j w_{ji} h_i \quad (1)$$

Where in equation (1) that each layer undirected edge represents dependencies only between hidden unit $h = (h_1, h_2, \dots, h_n)$; $h_i \in \{0,1\}$ where $i = 1, 2, \dots, n$ and visible units $v = (v_1, v_2, \dots, v_k)$; $v_j \in \{0,1\}$ where $j = 1, 2, \dots, k$. The binary data and labels so they can also be used to model the joint distribution.

The term of w_{ji} is the recurrent weight between visible unit j and hidden units i in the symmetric interaction, b_j and c_i are respectively referred to bias term between connection of hidden and visible units. The structure itself assigns a probability to each connection vector between hidden and visible units. This probability can be defined as mathematical equation with the help of energy functions:

$$P(v, h) = \frac{\exp(-E(v, h))}{Z} \quad (2)$$

Where Z :

$$Z = \sum_{v, h} \exp(-E(v, h)) \quad (3)$$

where Z as normalization constant between hidden and visible units can be obtained by summing over all possible pairs of visible and hidden unit vectors.

According to Hinton's paper work [8] the probability of the network assigns to training images can be improved by adjusting the weights and the biases to lower the energy of that's image and improve the other energy of images.

Due to predictive the data from the model in the speech to spectrograms, so we considered converting the spectrogram into image to create the same dimensional. The predictive between real data and the prediction model distribution as a vector with respect to a weight can be computed as follows:

$$-\frac{\partial \log P(v)}{\partial \mathcal{W}_{ji}} = \langle v_j h_i \rangle_{real} - \langle v_j h_i \rangle_{predict} \quad (4)$$

In equation (3), $\langle v_j h_i \rangle$ means to denote the expectation under the distribution specified by the subscript that follows. With simple learning rate rule to perform the stochastic ascent in the log probability of the training data can be shown as follows:

$$\Delta \mathcal{W}_{ji} = \alpha (\langle v_j h_i \rangle_{real} - \langle v_j h_i \rangle_{predict}) \quad (5)$$

The learning rate can be described as α . The function of using learning rate in DBN to search for momentum in updating weight and biases.

In RBM theorem shown that there is no connection between hidden and visible units so the hidden units are independent given by visible units. Now given a selected image v as randomly, the binary units product h_i of each hidden unit i^{th} , is set to 1 where the probability of the image can be calculated as:

$$P(h_i = 1|v) = g\left(b_i + \sum_j v_j \mathcal{W}_{ij}\right) \quad (6)$$

As shown in equation (5) the derivate of $g(x)$ is a *logistic sigmoid function*. The sigmoid can be written as $g(x) = 1/(1 + \exp(-x))$. So that's equation can be summarized as hidden unit also can be written as;

$$P(v_j = 1|h) = g\left(c_j + \sum_i h_i \mathcal{W}_{ij}\right) \quad (7)$$

2.3.2 Contrastive Divergence

Once RBM is learned using *Contrastive Divergence* (CD) algorithm [19], the DBN is able to initialize the weights of feed forward back-propagation neural network then it is used for classification to predict the image model. RBM can be learnt better when the predictive model is used before the step sampling in Gibbs before collecting statistical step in learning rule but for the purposes of pre-training.

In many cases when we are facing in continuous rather than binary the Gaussian Bernoulli RBM should be used to approach the data distribution. Continuous data such as MFCC can be naturally modeled by linear variable with Gaussian and the RBM energy function has been modified to approach such as variable, so the GRBM can be shown as:

$$E(v, h) = \sum_j \frac{(v_j - b_j)^2}{2\sigma_j^2} - \sum_i c_i h_i - \sum_j \sum_i \frac{v_j}{\sigma_i} \mathcal{W}_{ji} h_i \quad (8)$$

Where σ_i is the standard deviation of the Gaussian for visible units in . Since the binary data in the hidden layer we used the conditional distribution to sample the state. In the CD we enabled to create the conditional distribution both of hidden and visible layer.

The equation can be written as:

$$P(h_i|v) = g\left(b_i + \sum_j \frac{v_j}{\sigma_j} w_{ij}\right) \quad (9)$$

$$P(v_j|h) = \mathcal{N}\left(b_i + \sigma_i \sum_i \frac{h_i}{\sigma_j} w_{ij}, 1\right) \quad (10)$$

Where $\mathcal{N}(\mu, \sigma^2)$ is defined as Gaussian normal distribution. For some distribution, the RBMs may be not achieving representation as efficient as unrestricted Boltzmann Machine. However, if the layers have enough number of units hidden layers any distribution can be represented with RBMs. In addition, the number of hidden layer units with weight and bias helps to improve the performance of DBN using the log-likelihood. Log-likelihood represents the gradient of $\log p(v; \theta)$ the weights for the RBM can update as follows;

$$\Delta w_{ji} = E_{real}\langle v_j h_i \rangle - E_{predict}\langle v_j h_i \rangle \quad (11)$$

Where $E_{real}\langle v_j h_i \rangle$ representative of observed data in training set and $E_{predict}\langle v_j h_i \rangle$ is the prediction from the model.

RBM basically can be applied in speech recognition system. Using a RBM as a standard conversion model between speaker and speech spectral envelop, it is possible to recognize speech [31]. In this paperwork, we try to reconstruct the speech signal using Gaussian Mixer Model (GMM) as representatives from Hidden Markov Model (HMM) based voice conversion methods that are benefited from use of multiple RBM. The voiced subspace is used to train RBM in spectrogram perform feature.

2.4 Enhancing Feature Structure

This section describes a structure of optimization process by neural networks theorem. Some researcher focus on this works called Elman Network with a feedback layer which only connects to the hidden layer and they proposed the structural optimization to find the optimum characteristic parameters stated by Delgado et al [9]. After training the binary data in RBM, data from previous process can be used for training another model of first RBM significantly in hidden units this process can be repeated as much as desired for creating many layers of non-linear feature detectors and more complex data. The RBM stack can be associated in multi-layer generative model is called deep belief net [10].

This paper proposes enhancing method with setting parameters of each hidden layer and unit number of each layer in RBMs following by DBN.

This approaching method includes local search based on taboo search for structural optimization and the modularization based on solution space improves learning efficiency on calculation time. Here, the DBN structure optimization that the paper purposed can be described as follows:

Step 1: Optimization of number of hidden layer.

- Step 1-1: Set the n be the number of hidden layers and let $n = \underline{n}$. Let setting number of units in initial layer is 500. Using equation 6 to define probabilities from visible unit and hidden units. learning and verification are performed, the error at that time is ε_{st} , ε_{ve} respectively. Evaluation value of structure in DBN(E_n) is calculated from ε_{st} and ε_{ve} as the evaluation and classification value.

$$E_n = \frac{1}{\varepsilon_{st} + \varepsilon_{ve}} \quad (12)$$

- Step 1-2: When $E_n \geq E_n^*$ then update the best solution $E_n^* = E_n$, where $n^* = n$
- Step 1-3: if $\bar{n} > n$ then let $n = n+1$ and go to step 1-1, if $\bar{n} = n$, let n^* be the number of hidden layers.

Step 2: Optimize number of units of each layer using split the hidden layer with optimized solution space. The number of units is determined with the number of hidden layers. The number of units of hidden layer let be $(x_1, x_2, x_3, \dots, x_n)$.

- Step 2.1: Searching unit number of i^{th} hidden layer using the multilayer perceptron.
- Step 2.2: Let the center of gravity of subspace j be $x_1^j, x_2^j, \dots, x_n^j$ and let $\hat{x}_i^j = x_i^j$ as the current solution.
- Step 2.3: Let E_i^j calculated from equation above from the learning error ε_{st} and the verification error ε_{ve} of DBN structure for the representative point and used it as the evaluation value.
- Step 2.4: The highest value is selected and its set as D_t
- Step 2.5: Continue to search the highest value by repeating from step 2.1.

Step 3: Structure determination by optimizing the number of hidden layer units by taboo search.

- Step 3.1: Generate an solution $A_0 = (x_1, x_2, \dots, x_n)$ is chosen randomly from solution space D_t , set $A^* = A_0$ and made the axis of neighbor search: calculate the evaluation value at initial solution and set A_0 as the taboo list save to $t = 0$.
- Step 3.2: Evaluate each neighborhood solution by equation (11) as a set of neighboring solutions excluding the solutions included in the taboo list among neighborhood solutions of A^* learning. The most evaluated value A' be the high solution, then E_{tb} be the evaluation value and store it as A' in the taboo list.
- Step 3.3: When $E_{tb} > E_{tb}^*$, update $E_{tb}^* = E_{tb}$ to $A^* = A'$.

Step 3.4: If $t < T_{tb}$ go to step 3.2 with $t=t+1$. Otherwise if $t = T_{tb}$ the solution is finished with A^* as a solution. In this step, the structural optimization cannot decrease and seems monotonically increases.

3 Speech Recognition Using DBN

3.1 Experiment Preparation

In this section we describe the general idea in our system and also evaluate the DBN that has modularization system in simple dataset of voices. We try to process the speech signal using the traditionally simple voice dataset. That simple voice dataset consists of acoustic signal and expected containing meaningful of sound in the human range from 20 Hz to 20,000 Hz [12]. In the simple dataset that we chose shows the simple utterance experiment tried to test our system whether working well or not. We do not pick up the long of utterance due to our limitation time in study.

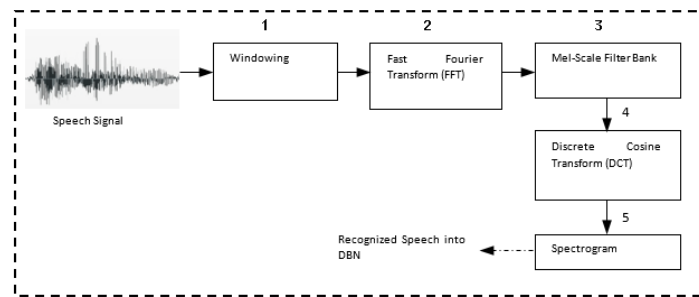


Figure 2. Proposed Model in Speech Recognition

3.1.1 Pre-Processing for Speech Recognition

In speech recognition there are so many techniques to store in traditionally of voices signal such as .mp3, .wav, .mid, etc. Despite of that in the pre-processing we traditionally chose the voices signal into .wav files for more comfortability. Then, in the feature extraction stage signal speech scripts converted into vector then combine it with MFCC feature of the signal as audio script to perform the input for the classifier.

Next process after we know the feature of audio script the machine learns to classify the recognized speech data. Our proposed method consists of several steps to extract the analog signal and digitalized the feature. The several steps extracted feature of the signal from proposed model in Figure 2 can be introduced as shown below:

1. Windowing

Speech is commonly dynamic time series signal in which the composition of properties changes very quickly over time. Before extracting the speech signal from analog to digital, at this stage we do frame blocking, the speech signal is divided into several frames with a general length of 20-30 ms containing N samples of each frame separated by M ($M < N$) where M is the number of shifts between frames. The first frame contains the first N sample. The second frame begins the sample M after the start of the first frame, so this second frame overlaps the first frame as much as the $N-M$ sample. Frame blocking is necessary because the voice signal changes over a period of time. The $N-M$ frames blocking can be illustrated as Figure 3.

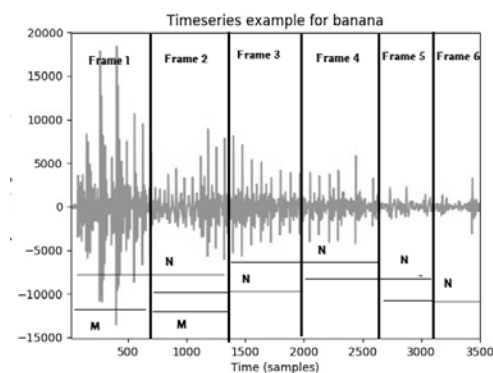


Figure 3 Frame Blocking Process

In the windowing process to minimize the discontinuity that occurs in the signal, which is caused by spectral leakage when the frame blocking process is done where the new signal, has a different frequency

with the original signal. The concept of windowing is to taper the end of the signal to zero at the beginning and end of each frame. By using windowing functions, the ability of an FFT to extract spectral data from signals can further enhance. Windowing functions act on raw data to reduce the effects of the leakage that occurs during an FFT of the data. The windowing process is multiplying each frame from the type of window used.

By designing the analog signal with hamming windows, the spectral analysis can be better for FFT processing. We use hamming windows to detect the peak of signal characteristic. The Hamming window has the lowest first side lobe level of all three types of windows. The slow decay means that leakage two or three bins away from a signal's center frequency are lower for the Hamming window.

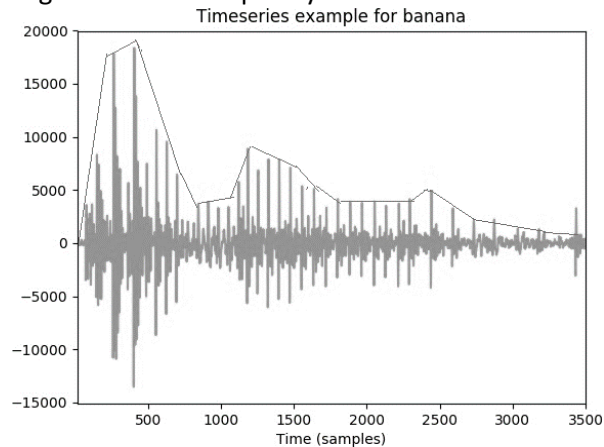


Figure 4 Hamming Window Process Example from Analog Signal

2. Fast Fourier Transform (FFT)

For better understanding, we choose the FFT due to the potential characteristic of signal waveform when detects the pattern in voices. Fast Fourier Transform algorithm serves as a signal modifier from time domain to frequency domain. The frequency values obtained from this process will be used in the filtering stage to obtain vector coefficients. Fast Fourier Transform (FFT) is a step to change each frame consisting of N samples from time domain into frequency domain. FFT is done to get the frequency of each frame. The output of this FFT process is a spectrum or periodogram.

One of the most popular FFT algorithms is radix-2. As a comparison with the DFT, for a large number of N samples such as $N = 512$, using DFT calculations requires a calculation of 114 times more than is required by FFT calculations. The larger the number of N samples, the more complex the calculation is if using DFT. The first step to interpret FFT is to calculate the frequency value of each middle sample of the FFT. If the sample time received by FFT is in real form, then only the output $X(m)$ from $m = 2$ to $m = N/2$ as independent. In this research, we calculate the FFT frequency value for $0 \leq m \leq N/2$ if the time sample received is complex, we must calculate all FFT frequency values of m from $0 \leq m \leq N - 1$.

3. Mel-Scale Filter

Mel-Frequency Wrapping uses Filter bank to filter the sound signals that have been converted into frequency domain form. Filter bank is a system that divides the signal input into a set of signal analysis, each corresponding to a different region spectrum. The performance of the MFCC is also influenced by one being the number of filters, in a study conducted by Vibha Tiwari in 2010 [58], the researchers used

a filter count of 32 filters, and resulted in better accuracy. The number of filters that are too many or too little will produce poor accuracy.

At this stage we will filter the signal for each frame. The filter used at this stage using filter bank mel. To make a filter bank mel, first set the upper and lower limits of the frequency. For the specified lower limit is 300 Hz and the upper limit is Sampling Frequency / 2 which is 8000 Hz.

4. Discrete Cosine Transform (DCT)

In this last stage, the value of mel will be converted back into time domain, the result is called Mel Frequency Cepstral Coefficient. This conversion is done using Discrete Cosine Transform (DCT). The average value in dB that can be used to estimate the energy coming from the filter bank. The DCT coefficient is the amplitude value of the resulting spectrum. At this stage the number of cepstrum taken is as much as 13 pieces per frame.

5. Spectrogram

After all process is completed the analog signal will be converted into spectrogram. A sound spectrograph (or sonogram) is a visual representation of an acoustic signal. To simplify things with a reasonable amount, the Fast Fourier transform is applied to electronically recorded sounds. Basically, this analysis separates the frequency and amplitudes of the simplex wave components. The results can be visually displayed from this spectrograph, we can see with the amplitude level (represented light to dark, like white = no energy, black = lots of energy), at various frequencies (usually on the vertical axis) by time (horizontal). For overall process in this research could be determined by figure 5.

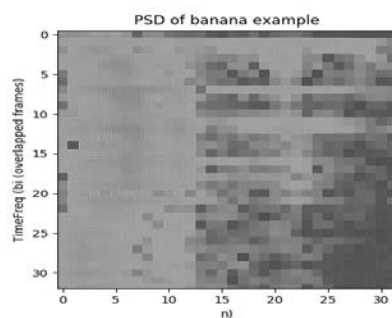


Figure 5 Spectrogram of Banana Example

3.1.2 Data Classification

The final step in our purposed model was classified and search the final accuracy of speech. A deep belief network is obtained by stacked several layers in RBM on top each other. The input of hidden layer at layer 'j' in our system becomes the input of the RBM layer at 'j+1'. The first layer in RBM as an input of network then the last of hidden layer in RBM represented the output. When used in classification, the DBN treated as Multi-Layer Perceptron (MLP). We used logistic regression as projecting data points onto a set of a hyperplane and calculated the distance to which is used to determine a class of membership probability.

3.2 Experimental Design

In our experimental design the simple dataset was traditionally recorded from Google Code Archive [17], this data set is consisting of 105 audio files in .wav formatted and each files containing utterance of one

fruit name spoken by a single speaker. These audio files are divided into seven class categories of fruit names i.e (apple, banana, kiwi, lime, orange, peach and pineapple) one category consists of 15 audio files. The whole dataset is separated into training (91 samples of audio) and testing (14 samples of audio). The detailed information shown in Table 2.

Table 1. Number of Data

Dataset	Apple	Banana	Kiwi	Lime	Orange	Peach	Pineapple
Number of train	14	14	14	14	14	14	14
Total Training	91						
Validation	105						

We randomly chose the data for training, validation and testing sets with ratio of 2:1:1 and did the observation of MFCC as well as for feature learned to test our system performances. After that, the voice signal analyzed with processing in windowing and fixed frame rate. Then processed to Fourier transform based on the log filter bank and the energy was disturbed in a Mel-scale, from this process the signal transformed into Discrete Cosine Transform (DCT) derived into MFCC features. Then, data were normalized so that each coefficient had zero mean and unit variance across the training into RBM unit layers.

4 Experiment and Discussion

4.1 Experiment Result

4.1.1 Pre-Processing for Speech Recognition Result

This section describes the experimental design in speech audio files. First step we are conducted the signal it could be read in our system, defined the signal voice with maximum size of 32 KHz then processed the signal into short-time Fourier transform (STFT) by windowing process using hamming-window (change the figure become time vs freq) as shown as figure 5 to reduce spectral leakage caused by the framing of the signal. After that signal waveform will be extracted into a single data matrix, and a label vector with the correct label for each data file is created.

Once of data has been inputted into a system and converted into a data matrix. The next step extracted the feature selection from the raw data, when it is done we have conducted the signal into Mel Frequency Cepstral Coefficient (MFCC). In this research, we used Short Time Fourier Transform (STFT) to approach the signal peak for processing into signal digital then converted it into the spectrogram.

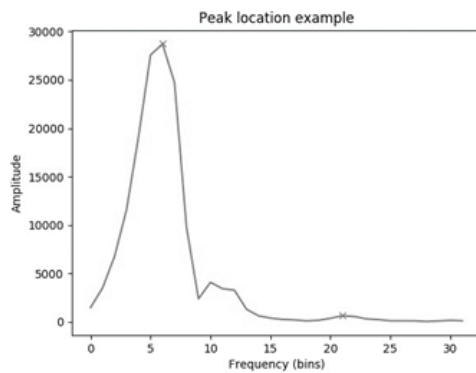


Figure 6. Spoken Word of Banana Time series

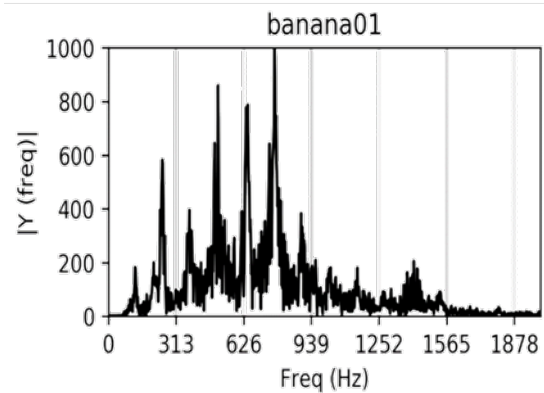


Figure 7. Peak Detection Spoken Word of Banana

Once we found the peak of the signal as seems like figure 6, the signal converted to “mfcc” vector and having six features. These mfcc features represented applying Gaussian Mixture Model (GMM). Feature extraction provides a complexity representation of digitalizing from speech. This digitalize perform Figure 7. indicates that spectral look of voice from sample speaks about (“apple”) utterance. This signal has characteristic recorded in 3.5 second and the frequency domain of speech is 16 KHz.

4.1.2 Data Classification Using DBN Result

In this section our dataset in speech recognition contains 367.5 seconds of speech. This number comes from in each single speaker the datasets consists of 3,5 second length of time then we time this with 105 total of dataset. The acoustic model training set was also used to train only single language for these experiments. Once the data has been inputted and turned into an input matrix, the next step is to extract feature from the raw data, then extracts the raw data into matrix the information of data input describe the sound over both frequency and time.

After that’s, we prepare the speech analyses using hamming window with fixed frames rate. In the Mel Frequency Cepstral Coefficient we normally use Cepstral mean normalization over each utterance. These are generated by applying a truncated discrete cosine transformation (DCT) to a log spectral estimate computed by smoothing a Fast Fourier Transforms (FFT) with around 20 frequency bins distributed across the speech spectrum to find peaks in frequency. Typically, we use 13 mfcc coefficients in our experiment.

The spectrogram in Figure. 7 has 30x30 dimensional matrix. This dimensional is advantageous for us to process in DBN through to RBM layer as binary data. For the first layer we used binary data to the RBMs input layer. We normalize the data so it has the zero mean and unit variance. Before processing into RBM the MFCC features attempt to eliminate the information of speech data when it is not having relevancies for recognition purpose. MFCC itself offered the alternative model as individual component to be independent so they are much easier to model using a mixture of diagonal covariance Gaussians.

We carried out the classification with a matrix to show that our system works well predicting the accuracy of each class in speech. This matrix result is obtained from the process of waveform in MFCC procedure then converted into RBM, from RBM the feature of MFCC combined onto multi-layer perceptron that processed in DBN. In Figure 6 indicates that the diagonally blocks our systems can predict the single word in 100% accuracy. As we can see each “block” from the figure shows the example predictive word “ap” means an apple, “ba” has a meaning of banana, also “ki” means kiwi and so forth.

4.2 Enhancing Result

In our experiment the signal is sampled in a range 8000 Hz and quantized with 16 bits. The signal is splits up in short frames of 80 samples corresponding to 10 ms of speech. That's range was choosing by relatively limited flexibility of the throat. When process into deep belief network we pick put the features from frequency domain and taking it with fast Fourier transform multiply by a hamming window to reduce the spectral leakage caused by framing of the signal. The input in DBN forming a total of 216 (6x36). So after we gets the binary data through to RBM, the structure of RBM has a weakness such as repeating learning could be consuming the time of structure of evaluation and leads to inefficiency in the structure optimization, so by the modularization structure of optimization is performed efficiently by shortening calculation time. By optimized the structure of DBN, we were able to find appropriate structure.

We consider using taboo search in partially of solution space and divided it into multiple subspaces first then the promising regions performed the search procedure. We conducted the experiment with 10 trial of each experiment with different modularization. In each trial, number of units hidden layers are similar to each other. Also we conduct of experiment with the number of hidden layer using different parameter setting, it's like 500-1000 and 2000 in single run of epochs.

Table 3. Performance Accuracy

Test Data	DBN (%)	DBN Structure Opt (%)
Speech	99.38	100

In this experiment we believe, using more number of unit in DBN, the feature is not good enough to generate the classifier. Due to the simple speech that we have conducted the performance is quite well enough. In the table 4, we tried to figure it out about the effect of varying size number of unit in initial space search solution. The main trend in table 4 is that adding more initial number of unit gives the better performance. Although, this research does not try the bigger size of dataset. In other hand, we tried an experimental design in HMM theorem for benchmarking algorithm, we got the accuracy of classification 80 % also with the same data set.

5 Conclusion and Future Work

This study review of the work in DBN and DBN improvement with RBM modularization as better as predicted the simplest speech audio signal files in better way accuracy. The modularization of hidden layer using taboo search is almost the same performance as DBN as without modularization. Even we set the minimum parameter setting of hidden layer size in 500, the performance is optimized well. Otherwise the speed of running the model much be increased when we were running the big data of speech.

The larger the number of solution dimensions the execution time will be shortened and made the effective of modularization. Then, the larger number of input dimensions the smaller of calculations amount in solution search. However, when comparing the structure optimized DBN without using DBN structurally optimized using modularization the performance is almost same.

In the future work, we will conduct some kind of the experiments on different voices dataset for benchmarking. Also, we will consider about different processing procedure in signal audio to improve the accuracy. Also we considered about speech in Parkinson diseases would be interested area.

REFERENCES

- [1] X.Huang, A. Acero and H.-W. Hon Spoken Language Processing: A Guide to Theory, Algorithm and System Development, Prentice Hall PTR, 2001.
- [2] Utpal B. C, "A Comparative study of LPCC and MFCC features for the recogniton of assamese phonemes", International Journal of Engineering Research and Technology (IJERT), 2013.
- [3] S. Shabani and Y. Norouzi, " Speech recognition using Principal Component Analysis and Neural Network," 2016 IEEE 8th International Conference on Intellegent Systems (IS), Sofia, pp.90-95, 2016.
- [4] S. S. Bhabad and G.K. Kharate, "Overview of technical progress in speech recognition", International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), 2013.
- [5] Mohamed, G. Dahl and G. Hinton, "Acoustic modeling using deep belief networks" *IEEE Transactions on Audio, Speech, and Language Processing*, 2011.
- [6] Ranganathan, H, Chakraborty, S. Panchanathan, *Multimodal emotion recognition using deep learning architectures*. in 2016 IEEE Winter Conference on Applications of Computer Vision, WACV 2016.
- [7] W. L., Zheng, JY ZHU, Y Peng, BL. Lu, *EEG-based emotion classification using deep belief networks*, Multimedia and Expo (ICME), IEEE International Conference on 2014, pp 1-6, 2014.
- [8] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, V. Vanhoucke, P. Nguyen, T. Sainath, B. Kingsbury, Deep neural networks for acoustic modeling in speech recognition, *IEEE Signal Process. Mag.* 29 (6) 82–97, 2012.
- [9] Y. Ishikawa, T. Hayashida, I. Nishizaki, S. Sekizaki, Improvement of structure optimization method of deep belief network, 2017 IEEE SMC Hiroshima Chapter Young Researchers Workshop, pp 56-60, 2017. (in Japanesse).
- [10] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley and Y. Bengio. "Theano: A CPU and GPU Math Expression Compiler". *Proceedings of the Python for Scientific Computing Conference (SciPy) 2010. June 30 - July 3, Austin*.
- [11] K. Swersky, B. Chen, B. Marlin, and N. de Freitas, "A tutorial on stochastic approximation algorithms for training Restricted Boltzmann Machines and Deep Belief Nets," in *Information Theory and Applications Workshop (ITA)*, 2010.
- [12] M. A. Carreira-Perpinan and G. E. Hinton, "On contrastive divergence learning," in *Artificial Intelligence and Statistics*, 2005.
- [13] T. Tieleman and G. Hinton, "Using fast weights to improve persistent contrastive divergence," in *Proceedings of the 26th Annual International Conference on Machine Learning*, New York, NY, USA, 2009.
- [14] O. Breuleux, Y. Bengio, and P. Vincent, "Quickly Generating Representative Samples from an RBM-Derived Process," *Neural Computation*, 2011.

- [15] G. E. Hinton, "Learning multiple layers of representation," Trends in Cognitive Sciences, 2007.
- [16] Y. Bengio, "Learning Deep Architectures for AI," Found. Trends Mach. Learn, 2009.
- [17] T. Tieleman, "Training restricted Boltzmann machines using approximation to the likelihood gradient," in Proceedings of the 25th international conference on Machine learning, New York, NY, USA, 2008.
- [18] Y. Bengio, A. Courville, and P. Vincent, "Unsupervised Feature Learning and Deep Learning: A Review and New Perspectives," 2012.
- [19] M. A. Keyvanrad and M. M. Homayounpour, "Deep Belief Network Training Improvement Using Elite Samples Minimizing Free Energy," 2014.
- [20] Y. Bengio, N. Chapados, O. Delalleau, H. Larochelle, X. Saint-Mleux, C. Hudon, and J. Louradour, "Detonation Classification from Acoustic Signature with the Restricted Boltzmann Machine," 2012.
- [21] Smolensky, P. Information processing in dynamical systems: Foundations of harmony theory. In Rumelhart, D. E. and McClelland, J. L., editors, Parallel Distributed Processing, 1986.
- [22] M. A. Keyvanrad and M. M. Homayounpour, "Effective Sparsity Control in Deep Belief Networks using Normal Regularization Term," submitted to Neural Networks, 2015.
- [23] J. Martens, "Deep Learning via Hessian-free Optimization," 2010.
- [24] N Morgan, "Deep and wide: Multiple layers in automatic speech recognition," IEEE Transactions on Audio, Speech, and Language Processing, 2012.
- [25] Sivaram G. and H. Hermansky, "Sparse multilayer perceptron for phoneme recognition," IEEE Transactions on Audio, Speech, and Language Processing, 2012.
- [26] T. N. Sainath, B. Kingsbury, and B. Ramabhadran, "Auto-encoder bottleneck features using deep belief networks," 2012.
- [27] N. Morgan, Qifeng Zhu, A. Stolcke, K. Sonmez, S. Sivasdas, T. Shinozaki, M. Ostendorf, P. Jain, H. Hermansky, D. Ellis, G. Doddington, B. Chen, O. Cretin, H. Bourlard, and M. Athineos, "Pushing the envelope - aside [speech recognition]," Signal Processing Magazine, IEEE, 2005.
- [28] O. Vinyals and S.V. Ravuri, "Comparing multilayer perceptron to deep belief network tandem features for robust asr," in Proceedings of ICASSP, 2011.
- [29] D. Yu, S. Siniscalchi, L. Deng, and C. Lee, "Boosting attribute and phone estimation accuracies with deep neural networks for detectionbased speech recognition," 2012.
- [30] L. Deng and D. Sun, "A statistical approach to automatic speech recognition using the atomic speech units constructed from overlapping articulatory features," Journal of the Acoustical Society of America, 1994.

- [31] J. Sun and L. Deng, "An overlapping-feature based phonological model incorporating linguistic constraints: Applications to speech recognition," *Journal of the Acoustical Society of America*, 2002.
- [32] P.C. Woodland and D. Povey, "Large scale discriminative training of hidden markov models for speech recognition," *Computer Speech and Language*, 2002.
- [33] Penghua Li, Shunxing Zhang, Huizong Feng, Yuanyuan Li, "Speaker Identification using Spectrogram and Learning Vector Quantization", 2015
- [34] Kshamamayee Dash, Debananda Padhi, Bhoomika Panda, Sanghamitra Mohanty, "Speaker Identification using Mel Frequency Cepstral Coefficient and BPNN", 2012
- [35] Dave, Namrata, "Feature Extraction Methods LPC, PLP And MFCC in Speech Recognition", 2013
- [36] Zhizheng Wu, et.al, "Vulnerability evaluation of speaker verification under voice conversion spoofing: the effect of text constraints", 2013.
- [37] J. Baker, L. Deng, J. Glass, S. Khudanpur, Chin hui Lee, N. Morgan, and D. O'Shaughnessy, "Developments and directions in speech recognition and understanding, part 1, 2009.
- [38] S. Furui, *Digital Speech Processing, Synthesis, and Recognition*, Marcel Dekker, 2000
- [39] C. Plahl, T. N. Sainath, B. Ramabhadran, and D. Nahamoo, "Improved pre-training of deep belief networks using sparse encoding symmetric machines," 2012.
- [40] B. Hutchinson, L. Deng, and D. Yu, "A deep architecture with bilinear modeling of hidden representations: applications to phonetic recognition," 2012.
- [41] Q. V. Le, J. Ngiam, A. Coates, A. Lahiri, B. Prochnow, and A. Y. Ng, "On optimization methods for deep learning," 2011.
- [42] N Morgan, "Deep and wide: Multiple layers in automatic speech recognition," *IEEE Transactions on Audio*, 2012.
- [43] Sivaram G. and H. Hermansky, "Sparse multilayer perceptron for phoneme recognition," 2012.
- [44] T. N. Sainath, B. Kingsbury, and B. Ramabhadran, "Auto-encoder bottleneck features using deep belief networks," 2012.
- [45] H. Bourlard, H. Hermansky, N. Morgan, *Towards increasing speech recognition error rates*, 1996
- [46] O. Siohan, Y. Gong, J.-P. Haton, "Comparative experiments of several adaptation approaches to noisy speech recognition using stochastic trajectory models", 1996.
- [47] V. Deng, M. Aksmanovik, *Speaker independent phonetic classification using HMMs with mixtures of trend functions*, 1997.
- [48] Hetherington, *PocketSUMMIT: small-footprint continuous speech recognition*, 2007.

- [49] H. Lin, J. Bilmes, D. Vergyri, K. Kirchhoff, OOV detection by joint word/phone lattice alignment, in: IEEE Automatic Speech Recognition and Understanding Workshop, 2007
- [50] K. Truong, D. van Leeuwen, Automatic discrimination between laughter and speech, 2007.
- [51] J. Fiscus, J. Ajot, J. Garofolo, The Rich Transcription 2007 Meeting Recognition Evaluation, in: Joint Proceedings of Multimodal Technologies for Perception of Humans, 2007.
- [52] S. Tranter, D. Reynolds, An overview of speech diarization systems, 2006
- [53] Y.-F. Liao, Z.-H. Chen, Y.-T. Juang, Latent prosody analysis for robust speaker identification, 2007.
- [54] S. F. Rashid. Optical Character Recognition – A Combined ANN/HMM Approach , 2014
- [55] Mostafa Hydari, Mohammad Reza Karami, Ehsan Nadernejad, Speech Signals Enhancement Using LPC Analysis based on Inverse Fourier Method, 2009.
- [56] Hyunsin Park, Tetsuya Takiguchi, and Yasuo Ariki, Research Article Integrated Phoneme SubspaceMethod for Speech Feature Extraction, 2009
- [57] Sishizuka K.& Nakatani T.: A feature extraction method using subband based periodicity and aperiodicity decomposition with noise robust frontend processing for automatic speech recognition, 2006.
- [58] Tiwari, Vibha, “MFCC and Its Application in Speaker Recognition”. International Journal on Emerging Technologies, 2010.

Artificial Human Optimization – An Overview

Satish Gajawada

*Alumnus, Indian Institute of Technology Roorkee
Founder, Artificial Human Optimization – A New Field
gajawadasatish@gmail.com*

Hassan M. H. Mustafa

*Faculty of Specified Education, Dept. of Educational Technology, Banha University, Egypt
prof.dr.hassanmoustafa@gmail.com*

ABSTRACT

The main idea to author this article is to “Popularize Artificial Human Optimization Field like never before by showing an Overview of this new field”. This idea can be divided into following sub-ideas:

- 1) To show the definition of “Artificial Human Optimization Field (AHO Field)”
- 2) To show difficulty level of creating new algorithms under AHO field
- 3) To show 30+ titles of papers published under AHO field
- 4) To show names of 65+ authors who worked under AHO Field
- 5) To show best negative reviews obtained for work under AHO Field
- 6) To show best positive reviews obtained for work under AHO Field
- 7) To show feedback given by an expert for work under AHO Field
- 8) To show “Hassan Satish Particle Swarm Optimization (HSPSO)”. This is latest work under AHO Field
- 9) To show contribution of Satish Gajawada and co-authors to this new Field
- 10) To show surprising results obtained after implementing AHO algorithms
- 11) To show you “Future of Artificial Human Optimization Field”

1 Sub-idea 1: Definition

Artificial Human Optimization (AHO) is a new field proposed in December 2016. This work was published in Transactions on Machine Learning and Artificial Intelligence. All optimization methods which were proposed based on Artificial Humans will come under the new field titled Artificial Human Optimization [1]. The first paper in AHO field was proposed in 2006 [2].

2 Sub-idea 2: Difficulty Level

The following is the review obtained from an expert in 2013 for a work in AHO Field:

“The motivation of the paper is interesting. But the paper does not present any evaluation of the proposed algorithm. So we have an idea but we are not able to assess it on the basis of the paper. Next, there seems to be a difference between birds, fishes, ants, bacteria, bees etc. on one side, and human beings on the

other side. Birds, fishes, ants, bacteria, bees etc. are more or less the same. People are different. I dare say that taxi drivers are different from politicians, or preschool teachers for example. Some people prefer money or power than love. It is not so difficult to guess which way ants will go but it is not so obvious when we consider people behavior. In my opinion the paper is a very first step to build the algorithm assumed but still lots of work is needed to achieve the goal.”

From the above review it is clear that optimization methods based on Humans is not as easier as developing optimization methods based on Birds, fishes, ants, bacteria, bees etc.

3 Sub-idea 3: Titles of Papers

The following are the titles of papers published under AHO Field according to [3]:

- 1) Human behavior-based optimization: a novel metaheuristic approach to solve complex optimization problems
- 2) Human Behavior Algorithms for Highly Efficient Global Optimization
- 3) Human Behavior-Based Particle Swarm Optimization
- 4) POSTDOC : THE HUMAN OPTIMIZATION
- 5) Focus Group: An Optimization Algorithm Inspired by Human Behavior
- 6) ENTREPRENEUR : Artificial Human Optimization
- 7) Modification of particle swarm optimization with human simulated property
- 8) Human cognition inspired particle swarm optimization algorithm
- 9) Human-inspired algorithms for continuous function optimization
- 10) Anarchic Society Optimization: A human-inspired method
- 11) The Human-Inspired Algorithm: A Hybrid Nature-Inspired Approach to Optimizing Continuous Functions with Constraints
- 12) CEO: Different Reviews on PhD in Artificial Intelligence
- 13) Artificial Human Optimization – An Introduction
- 14) An Ocean of Opportunities in Artificial Human Optimization Field
- 15) 25 Reviews on Artificial Human Optimization Field for the First Time in Research Industry
- 16) A New Optimization Method Based on Adaptive Social Behavior: ASBO
- 17) Human meta-cognition inspired collaborative search algorithm for optimization
- 18) Self regulating particle swarm optimization algorithm
- 19) Improved SRPSO algorithm for solving CEC 2015 computationally expensive numerical optimization problems
- 20) Clustering of Text Document based on ASBO
- 21) PID Controller Auto tuning using ASBO Technique

- 22) ASBO Based Compositional in Combinatorial Catalyst
- 23) Seeker Optimization Algorithm
- 24) Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems
- 25) Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition
- 26) Group Counseling Optimization: A Novel Approach
- 27) A Simple Human Learning Optimization Algorithm
- 28) A novel optimization algorithm inspired by the creative thinking process
- 29) Immigrant Population Search Algorithm for Solving Constrained Optimization Problems
- 30) Democracy-inspired particle swarm optimizer with the concept of peer groups
- 31) Social Emotional Optimization Algorithm for Nonlinear Constrained Optimization Problems
- 32) Human opinion dynamics: An inspiration to solve complex optimization problems

4 Sub-idea 4: Names of authors

The following are the researchers who made their contribution to AHO Field according to [3]:

- 1) Satish Gajawada
- 2) Hassan M. H. Mustafa
- 3) Seyed-Alireza Ahmadi
- 4) Da-Zheng Feng
- 5) Han-Zhe Feng
- 6) Hai-Qin Zhang
- 7) Hao Liu
- 8) Gang Xu
- 9) Gui-yan Ding
- 10) Yu-bo Sun
- 11) Edris Fattahi
- 12) Mahdi Bidar
- 13) Hamidreza Rashidy Kanan
- 14) Ruo-Li Tang
- 15) Yan-Jun Fang
- 16) Muhammad Rizwan Tanweer

- 17) Suresh Sundaram
- 18) L. M. Zhang
- 19) C. Dahlmann
- 20) Y. Zhang
- 21) A. Ahmadi-Javid
- 22) Mingyi Zhang, Luna
- 23) Zhang, Yanqing
- 24) Manoj Kumar Singh
- 25) N. Sundararajan
- 26) Prakasha S
- 27) H R Shashidhar
- 28) G T Raju
- 29) Sridhar N
- 30) Nagaraj Ramrao
- 31) Manoj Kumar Singh
- 32) Devika P. D
- 33) Dinesh P. A
- 34) Rama Krishna Prasad
- 35) Chaohua Dai
- 36) Yunfang Zhu
- 37) Weirong Chen
- 38) R.V.Rao
- 39) V.J.Savsani
- 40) D.P.Vakharia
- 41) Esmaeil Atashpaz-Gargari
- 42) Caro Lucas
- 43) M. A. Eita
- 44) M. M. Fahmy
- 45) Ling Wang
- 46) Haoqi Ni

- 47) Ruixin Yang
- 48) Minrui Fei
- 49) Wei Ye
- 50) Xiang Feng
- 51) Ru Zou
- 52) Huiqun Yu
- 53) Hamid Reza Kamali
- 54) Ahmad Sadegheih
- 55) Mohammad Ali Vahdat-Zad
- 56) Hassan Khademi-Zare
- 57) Ritambhar Burman
- 58) Soumyadeep Chakrabarti
- 59) Swagatam Das
- 60) Yuechun Xu
- 61) Zhihua Cui
- 62) Jianchao Zeng
- 63) Rishemjit Kaur
- 64) Ritesh Kumar
- 65) Amol P. Bhondekar
- 66) Pawan Kapur

5 Sub-idea 5: Best Negative Reviews

The following are the best negative reviews obtained for work under AHO field:

- 1) This paper studies a so-called human optimization method which falls into the research topic of optimization. The proposed method was presented on the first page followed by some discussions. The paper clearly makes no novel contribution to the state of the art on optimization algorithms and techniques. Thus, because of this lack of new contribution, the paper is not appropriate for the conference.
- 2) Nothing to evaluate
- 3) Funny paper, especially the notion of "love array" :)
- 4) This is not a research paper. It should not have been submitted for review. Rationale and results are completely lacking. I do not even think there is a research idea in there.

6 Sub-idea 6: Best Positive Reviews

The following are the best positive reviews obtained for work under AHO field:

- 1) We had a glance at your published article "POSTDOC : THE HUMAN OPTIMIZATION". We found your article very innovative, insightful and interesting. We really value your outstanding contribution towards Scientific Community.
- 2) Very Interesting (from IEEE TAAI 2013)
- 3) Very Novel (from Springer SOCTA 2017)
- 4) Very Impressive
- 5) Compelling and Creative (from experts of aitoday.xyz)
- 6) New and Interesting Area of Research (from world class conference PAKDD 2018)

7 Sub-idea 7: Feedback received by Satish Gajawada

Below is the feedback from an expert when Satish Gajawada (Founder, Artificial Human Optimization) asked to give feedback on work under AHO:

"Thanks for the message. It seems you are the "father of Artificial Human Optimization" field, it will be tomfoolery on my part to provide feedback on such topic. You are already at the zenith of this research."

8 Sub-idea 8: Latest Work

Hassan Satish Particle Swarm Optimization (HSPSO) is the latest work under AHO Field. It is shown below:

HSPSO is obtained by incorporation of MSHO concepts into Particle Swarm Optimization. In starting and even generations the Artificial Humans move towards the best fitness value. In odd generations Artificial Humans move away from the worst fitness value. In HSPSO, we maintain local worst of particle and global worst of all particles in addition to local best of particle and global best of all particles. This is shown in lines 4 to 17. In lines 19 to 24 velocity is calculated by moving towards the local best of particle and global best of all particles. In lines 26 to 31 pseudo code for odd generations is shown in below text. In these odd generations particles move away from local worst of particle and also away from global worst of all particles. In line 33, number of iterations is incremented by one. Then control goes back to line number 4. This process of moving towards the best in one generation and moving away from the worst in next generation is continued until termination criteria has been reached.

Procedure: Hassan Satish Particle Swarm Optimization (HSPSO)

- 1) Initialize all particles
- 2) iterations = 0
- 3) **do**
- 4) **for** each particle i **do**
- 5) **if** ($f(x_i) < f(pbest_i)$) **then**
- 6) $pbest_i = x_i$
- 7) **end if**
- 8) **if** ($f(pbest_i) < f(gbest)$) **then**
- 9) $gbest = pbest_i$
- 10) **end if**

```

11)         if ( f( xi ) > f( pworsti ) ) then
12)             pworsti = xi
13)         end if
14)         if ( f( pworsti ) > f( gworst ) ) then
15)             gworst = pworsti
16)         end if
17)     end for
18)     if ((iterations == 0) || (iterations%2==0)) then // for starting and even iterations
19)         for each particle i do
20)             for each dimension d do
21)                 vi,d = vi,d + C1*Random(0,1)*(pbesti,d - xi,d) + C2*Random(0,1)*(gbestd - xi,d)
22)                 xi,d = xi,d + vi,d
23)             end for
24)         end for
25)     else // for odd iterations
26)         for each particle i do
27)             for each dimension d do
28)                 vi,d = vi,d + C1*Random(0,1)*( xi,d - pworsti,d ) + C2*Random(0,1)*( xi,d - gworstd)
29)                 xi,d = xi,d + vi,d
30)             end for
31)         end for
32)     end if
33)     iterations = iterations + 1
34) while ( termination condition is false)

```

9 Sub-idea 9: Contribution of Satish Gajawada and Co-authors

- 1) Entrepreneur: Artificial Human Optimization. Transactions on Machine Learning and Artificial Intelligence, Volume 4 No 6 December (2016); pp: 64-70.
- 2) CEO: Different Reviews on PhD in Artificial Intelligence, Global Journal of Advanced Research, vol. 1, no.2, pp. 155-158, 2014.
- 3) POSTDOC : The Human Optimization, Computer Science & Information Technology (CS & IT), CSCP, pp. 183-187, 2013.
- 4) Artificial Human Optimization – An Introduction, Transactions on Machine Learning and Artificial Intelligence, Volume 6, No 2, pp: 1-9, April 2018.
- 5) An Ocean of Opportunities in Artificial Human Optimization Field, Transactions on Machine Learning and Artificial Intelligence, Volume 6, No 3, June 2018.
- 6) 25 Reviews on Artificial Human Optimization Field for the First Time in Research Industry, International Journal of Research Publications, Volume 5, No 2, United Kingdom, 2018.
- 7) Collection of Abstracts in Artificial Human Optimization Field, International Journal of Research Publications, Volume 7, No 1, United Kingdom, 2018.
- 8) HIDE : Human Inspired Differential Evolution - An Algorithm under Artificial Human Optimization Field, International Journal of Research Publications (Volume: 7, Issue: 1), http://ijrp.org/paper_detail/264
- 9) Hybridization concepts of Artificial Human Optimization field Algorithms incorporated into Particle Swarm Optimization (In Progress)
- 10) Artificial Human Optimization – An Overview

11) An Artificial Human Optimization Algorithm titled Human Thinking Particle Swarm Optimization (In Progress)

12) Testing Multiple Strategy Human Optimization based Artificial Human Optimization Algorithms (In Progress)

10 Sub-idea 10: Surprising Results

In [10], PSO method performed well than the proposed HPSO method for particular parameter settings and benchmark function. But the general expectation is that after adding Artificial Human Optimization concepts into PSO, the proposed HPSO method should perform well. But this is not the case.

11 Sub-idea 11: Future

According to [7], there exists millions of opportunities in AHO Field. Some interesting opportunities possible in near Future are shown below:

- 1) International Institute of Artificial Human Optimization, Hyderabad, INDIA
- 2) Indian Institute of Technology Roorkee Artificial Human Optimization Labs, IIT Roorkee
- 3) Foundation of Artificial Human Optimization, New York, USA.
- 4) IEEE Artificial Human Optimization Society
- 5) ELSEVIER journals in Artificial Human Optimization
- 6) Applied Artificial Human Optimization – A New Subject
- 7) Advanced Artificial Human Optimization – A New Course
- 8) Invited Speech on “Artificial Human Optimization” in world class Artificial Intelligence Conferences
- 9) A Special issue on “Artificial Human Optimization” in a Springer published Journal
- 10) A Seminar on “Recent Advances in Artificial Human Optimization” at Technical Festivals in colleges

REFERENCES

- [1] Satish Gajawada; Entrepreneur: Artificial Human Optimization. Transactions on Machine Learning and Artificial Intelligence, Volume 4 No 6 December (2016); pp: 64-70
- [2] Dai C., Zhu Y., Chen W. (2007) Seeker Optimization Algorithm. In: Wang Y., Cheung Y., Liu H. (eds). Computational Intelligence and Security. CIS 2006. Lecture Notes in Computer Science, vol 4456. Springer, Berlin, Heidelberg.
- [3] Satish Gajawada and Hassan M. H. Mustafa, “Collection of Abstracts in Artificial Human Optimization Field”, International Journal of Research Publications, Volume 7, No 1, United Kingdom, 2018.
- [4] Satish Gajawada, “CEO: Different Reviews on PhD in Artificial Intelligence”, Global Journal of Advanced Research, vol. 1, no.2, pp. 155-158, 2014.
- [5] Satish Gajawada, “POSTDOC : The Human Optimization”, Computer Science & Information Technology (CS & IT), CSCP, pp. 183-187, 2013.
- [6] Satish Gajawada, “Artificial Human Optimization – An Introduction”, Transactions on Machine Learning and Artificial Intelligence, Volume 6, No 2, pp: 1-9, April 2018.

- [7] Satish Gajawada, "An Ocean of Opportunities in Artificial Human Optimization Field", Transactions on Machine Learning and Artificial Intelligence, Volume 6, No 3, June 2018.
- [8] Satish Gajawada, "25 Reviews on Artificial Human Optimization Field for the First Time in Research Industry", International Journal of Research Publications, Volume 5, No 2, United Kingdom, 2018.
- [9] Satish Gajawada, Hassan M. H. Mustafa , HIDE : Human Inspired Differential Evolution - An Algorithm under Artificial Human Optimization Field , International Journal of Research Publications (Volume: 7, Issue: 1), http://ijrp.org/paper_detail/264
- [10] Satish Gajawada and Hassan M. H. Mustafa, Hybridization concepts of Artificial Human Optimization field Algorithms incorporated into Particle Swarm Optimization (In Progress)
- [11] Satish Gajawada and Hassan M. H. Mustafa, Artificial Human Optimization – An Overview.
- [12] Satish Gajawada and Hassan M. H. Mustafa, An Artificial Human Optimization Algorithm titled Human Thinking Particle Swarm Optimization (In Progress).
- [13] Satish Gajawada and Hassan M. H. Mustafa, Testing Multiple Strategy Human Optimization based Artificial Human Optimization Algorithms (In Progress).

Development of an Electronic Nose for Olfactory System Modelling using Artificial Neural Network

¹Mary Anne Roa, ²Proceso Fernandez

¹Department of Information Systems & Computer Science, Ateneo de Manila University, Philippines;
mary.roa@obf.ateneo.edu; pfernandez@ateneo.edu

ABSTRACT

Electronic nose (e-nose) devices have received considerable attention in the field of sensor technology because of their many potential uses such as in identification of toxic wastes, monitoring air quality, examining odors in infected wounds and in inspection of food. Notwithstanding the vast amount of literature on the usage of e-noses for specific purposes, the technology originally and ultimately aims to mimic the capability of mammals to discriminate odors from all sorts of objects. This study demonstrates the theoretical and practical feasibility of designing an e-nose towards general odor classification. A multi-sensor array hardware unit was carefully constructed for data collection and odor detection. Important hardware design considerations such as sensor calibration, aeration, circuit protection, and voltage/current requirements were satisfied. A highly fine-tuned artificial neural network (ANN) was integrated to the hardware to interpret and relate the data to a target odor class from a set of 10 primary odors identified in a previous study. Various network architecture considerations, such as neuron count, number of layers and activation function, as well as various data treatment methods, such as normalization, and data partitioning, were investigated. The results showed that careful hardware integration with an ANN having sufficiently deep internal structure can yield accurate classification to at least half of the ten primary odor classes, namely fragrant (96%), fruity (98%), chemical (99%), peppermint (98%), and popcorn (90%). The results demonstrate the feasibility of making e-noses for general odor classification, which could lead to further broadening of e-nose applications.

Keywords: Artificial Neural Network; Odor Classification; Electronic Nose; Machine Learning.

1 Introduction

Did you ever measure a smell? Can you tell whether one smell is just twice strong as another? Can you measure the difference between two kinds of smell and another? It is very obvious that we have very many different kinds of smells, all the way from the odor of violets and roses. But until you can measure their likeness and differences, you can have no science of odor. If you are ambitious to find a new science, measure a smell. — Alexander Graham Bell, 1914

The research is supported by the Department of Science and Technology - Engineering Research and Development for Technology, Philippines.

Scientists as well as philosophers have long neglected olfaction. The lack of interest in the nature of smells stems perhaps from its very character. As the most volatile sense of all, the smell does not appear to be

sufficiently real. Odors are considered as too insubstantial and very brief in appearance [1]. It thus does not come as a surprise that there is no complete understanding of how smell perception works yet, especially in humans [2]. The mechanism of the olfactory system has been studied but not as thoroughly as visual and auditory systems where excellent electrical analogues are available [3]. One of the initial hopes for work in this area was connecting distinct biological attributes with hardware, as well as capturing odor fingerprint, therefore creating an electronic nose (e-nose) [4].

Artificial olfaction can trace its beginnings with the invention of the first gas multi-sensor array in 1982 [5]. Recently, it has received considerable attention largely due to the discovery of numerous applications in diverse fields of applied sciences such as agricultural, biomedical, cosmetics, environmental, food, manufacturing, military, pharmaceutical, regulatory, and more. These include analysis of fuel mixtures [6], detection of oil leaks [7], identification of household odors [8], examining breath [9], analysis of body fluids [10, 11] etc. Due to these numerous applications and promising benefits to a diverse field, the global e-nose market has been forecasted to grow by 2.07% during the period 2016-2020 [12].

The technology of artificial olfactory system has continuously been advancing up to the present. However, although many e-nose models have been developed, the vast majority of these are intended for very specific applications. Making e-noses adept in categorizing smells of various samples is a big step towards broadening its application.

This study focused on the development of an e-nose that is directed towards general odor classification, by using the 10 primary smell classes suggested by researchers from University of Pittsburgh and Bates College [13]. Their study presented that odor dimensions apply categorically and that scents can be placed in one of ten basic categories of odor -- fragrant, woody, fruity, chemical, peppermint, sweet, popcorn, lemon, pungent, and decayed.

This study integrated different olfaction sensors in a compact e-nose prototype that can make fine discrimination on the chemical properties of odors from different objects. An Artificial Neural Network (ANN) was used to make reasonable decisions about the categories of the smell. Various ANN configurations and data treatment methods affecting overall system performance were investigated.

2 Methodology

2.1 Hardware Design Consideration

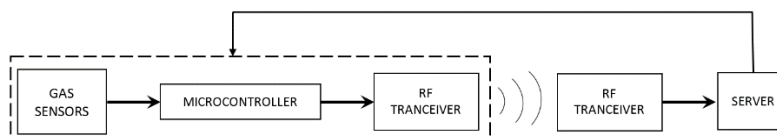


Figure 1 Overall System Block Diagram

Hardware design is the elaborate process to make a prototype to satisfy requirements. The process of designing e-nose generally begins with fulfilling the need for air to flow into the assembly to remove various types of gaseous and particulate contaminants that may affect or interfere with measurements. The sensor elements' calibration is equally vital. To design the auxiliary circuit, current specification, voltage specification and transient protections were considered to meet the specific functional requirements on power regulation. Figure 1 show the overall system block diagram.

2.1.1 Sensor Integration

An e-nose fundamentally consists of three major parts, and these are the sample delivery system, detecting system, and processing system. Detection typically uses an array of sensors of different sensitivities that simultaneously respond to the volatile chemicals present in a sample. The pattern of response for different odorants is important, as this distinguishability allows the system to identify an odor. In this study, the prototype was realized by interfacing 10 semiconductor gas sensors, namely MQ2 for smoke, MQ3 for alcohol, MQ4 for natural gas, MQ5 for LPG, MQ6 for butane and propane, MQ7 for carbon monoxide, MQ8 for hydrogen gas, MQ9 for methane, MQ135 for benzene and MQ138 for ammonia. The sensors send measurements to the server through a ZigBee transmitter. The configuration of the MQ gas sensor is shown in Figure II.



Figure 2. Sensor (a) front view, (b) top view, (c) module and (c) pin configuration

2.1.2 Aeration

The sample delivery system is essential to guarantee stable operating conditions and to avoid contamination of samples from external odor sources. Insulation or sealing materials that may give off fumes such as silicone sealants and solvent-based adhesives were avoided. Gas is supplied to detectors using a static gas mixture, thus avoiding direct gas flow onto the sensor that can cause a false high reading. Ambient air can pass into the sampling chamber after every gas exposure cycle using an axial flow fan that draws air into the case from the outside and expels a previous sample's gases from the inside.

2.1.3 Auxiliary Requirements:

A schematic diagram of the power supply used is shown in Figure III. A voltage regulator maintains the sensor output voltage level requirement of 5 VDC at point (a). Another integrated circuit at point (b) was used to further regulate the output to 3.3V in order to satisfy the requirements of the ZigBee module and the microcontroller. The microcontroller used, described in Figure IV, serves as the controller for the different sensors and auxiliary components of the system.

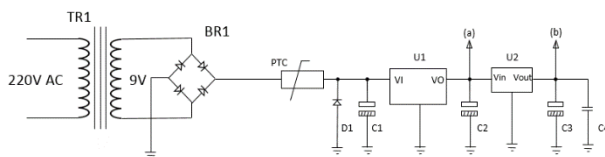


Figure 3. Main Supply Schematic Diagram

Figure V depicts a scale model of the assembly. The bigger block serves as the sample delivery chamber where an object being analyzed is placed and into which odorless air is ducted after sampling. The adjacent smaller block houses the auxiliary electronics.

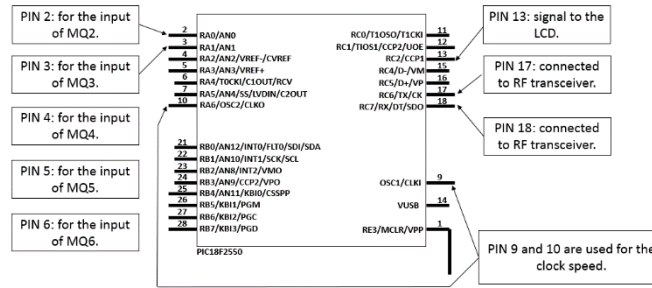


Figure 4. Microcontroller Unit Connection Diagram

The output category of smell is displayed on an LCD monitor integrated to the e-nose prototype. Options to “classify” and “clear” allow a user to trigger the device to either read and process samples or turn the exhaust for clearing gas out of the chamber.

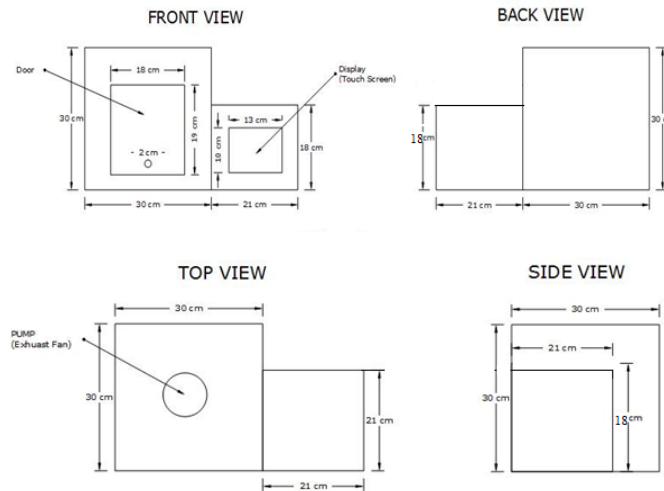


Figure 5. Orthographic Projection of the Prototype

2.2 Software Configuration

An artificial neural network (ANN) interprets the multi-dimensional data yield by the hardware unit and relates this to a target odor class. The ability of an ANN to accurately classify the odor sample depends a lot on both its model parameters, i.e., network connection weights, and on its hyperparameters that include the number of layers, the number of neurons in each layer and the activation function used in a neuron. The network parameters are typically set through training the network using some form of gradient descent learning algorithm combined with back propagation. The hyperparameters, on the other hand, are typically handcrafted. To achieve good accuracy, extensive experimentation was conducted on various ANN hyperparameters, as discussed in the succeeding paragraphs. Figure VI hints on the overall iterative nature in the design of the final ANN used in the e-nose for this study.

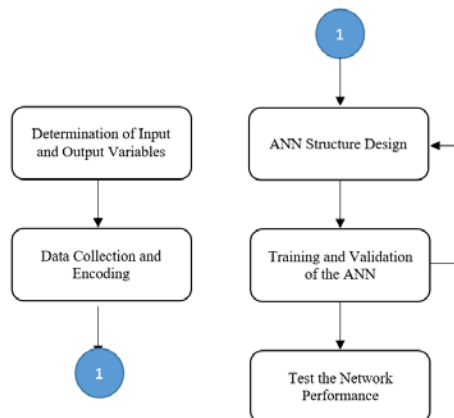


Figure 6. Artificial Neural Network Design Workflow

2.2.1 Neuron Count

The neurons in the hidden layer are responsible for the internal representation of the data and the information transformation between input and output layers. A preliminary study [14] on the effect of neuron count on odor classification showed that small networks were unable to model the odor pattern. An increase in the number of neurons in the hidden layers resulted in better fit. However, increasing the number of neurons beyond some threshold has an opposite effect when verified against the test data. This is because having more hidden neurons means that there are a lot more model parameter to be learned, and the corresponding much larger (optimization) search space makes it more difficult to find the optimal parameters.

2.2.2 Number of Layers

As ANNs can consist of multiple layers of computational units, various network structures were also trained and tested in order to find out the relationship between the number of hidden layers against the overall classification performance. Results from a previous work [15] showed that the structure of the network has a perceptible impact on learning time. Evidently, large networks took longer time to learn the characteristics of the data. However, an ANN with small internal structure did not give good approximations even for patterns included in its training set. Increasing the number of layers improved generalization of the network. In this paper, we denote the network topology by a sequence of m numbers that describe the sizes of the m layers. For instance, 10-3-9-10 describes a network with 10 neurons in the first layer, 3 in the second layer, 9 in the third layer, and 10 neurons in the last layer.

2.2.3 Activation Function

The activation function controls the total signal that a neuron produces given its input. It affects the power of a neural network in several aspects such as efficiency of weight updates, the speed of learning and final network complexity. Quantitative comparisons of four of the most commonly used ANN activation functions were explored in [16]. Networks with linear activation functions are consistently fastest, followed closely by those with Rectified Linear Unit (ReLU). The sigmoid and hyperbolic tangent activation functions involve more costly computations and are thus significantly slower, but they generally yield the most accurate results. However, the difference in the accuracies between the ReLU and the sigmoid or

hyperbolic tangent is generally not significant when a properly trained network has sufficient number of hidden layers and neurons in each layer. The faster learning capability with the use of ReLU, when compared with the other non-linear activation functions, thus makes it the preferable activation function for large models trained on large datasets.





Activation Function	Mathematical Equation	2D Graphical Representation	Range
Linear	$f(x) = x$		$(-\infty, \infty)$
Sigmoid	$f(x) = \frac{1}{1 + e^{-x}}$		$(0, 1)$
Hyperbolic Tangent	$f(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$		$(-1, 1)$
ReLU	$f(x) = \max(0, x)$		$(0, \infty)$

Table 1. Activation Functions

2.3 Data Collection and Treatment

A big challenge in e-nose design, besides hardware and software design considerations, is getting and processing the right data. ANN design is generally better when the number of training samples is substantial, as this allows for more free parameters to be incorporated in the model. The sample data collection was done carefully by ensuring the proper working conditions of the e-nose hardware unit. Any traces of smoke, adhesives, gases, or solvents were removed from the chamber, and the atmospheric conditions in the chamber were maintained stable at room temperature and humidity.

A series of measurements was captured for each sample. Each measurement comprised of an exposure cycle followed by a cleaning phase where the ambient air was passed into the chamber through the axial flow fan. The exposure was fixed at 15 seconds to provide enough time for a quantifiable response from all the sensors. The cleaning phase was set to 15 seconds to permit an acceptable recovery of the sensors sensitive to the sampled material and ensure that all gases or fumes are expelled from inside the chamber.

During the sample data collection, five measurements for each sensor were obtained and sent to the server for processing. This was repeated 3 times per sample. Thus, a raw data matrix with dimension 15x10 for each sample was obtained. The 10 sensor measurements, 15 collected readings per sample, and 100 different sample materials collectively yielded a 1500x10 (readings by feature/sensor) dataset that can be downloaded from [17]. The dataset is balanced, with each odor class having 150 sample readings based on 10 different sample materials per odor class.

The collected dataset was used in different ways to benchmark candidate ANNs under various experiment configurations. Specifically, the dataset was used in investigating the effect on the ANN performance of data pretreatment through normalization, in exploring for a good partitioning of the normalized dataset, and then in searching for the ANN hyperparameters and parameters using the 10-fold validation results as estimates of ANN performance.

2.3.1 Data Normalization

Working directly on raw data with arbitrary units and different magnitudes could affect the learning algorithm negatively because the features with bigger values may dominate those with smaller ones. Normalization of the input data is usually carried out in order to address this issue.

In this study, the hyperparameter search was performed on raw and normalized data (min-max and z-score). The results, in terms of precision (P), recall (R) and f-measure (f), are summarized in Table II and clearly indicate the benefit of performing data normalization. Aside from producing better scores, training the ANN on normalized data, also allowed relatively smaller networks to achieve accurate classification.

Table 2. Hyperparameter Tuning Results with Normalization

Data Treatment	ANN	P	R	f
Raw	10-10-6-8-10	0.61	0.61	0.60
	10-8-6-10-10	0.68	0.68	0.67
	10-8-4-8-10	0.74	0.76	0.73
	10-9-10-10	0.76	0.78	0.77
Min-max scaled	10-7-2-3-10	0.73	0.75	0.74
	10-10-7-10	0.80	0.79	0.79
	10-8-5-10	0.83	0.82	0.82
	10-3-9-10	0.83	0.82	0.82
Z-standardized	10-5-4-3-10	0.72	0.72	0.72
	10-6-3-5-10	0.74	0.76	0.75
	10-9-8-10	0.79	0.80	0.79
	10-8-9-10	0.84	0.84	0.84

2.3.2 Dataset Partitioning

A general practice for improving generalization when training an ANN is to first divide the available data into three subsets -- training, validation and test sets. This partitioning is called the holdout method. To investigate the impact on the ANN model performance of the proportion of data in the holdout method, ANN models from hyperparameter tuning were trained on a partitioned min-max normalized dataset, and errors on the test subsets were observed. Results in Table III show that the best overall result was obtained when 80% of the data is used for training, 10% for validation and 10% for testing. It is not a recommended approach to cut the test subset excessively, as in 80-15-5, as it affects negatively the generalizing power of an ANN. On the other hand, leaving no data for validation, as in 90-0-10, makes overfitting much more likely to happen.

Table 3. Result of Holdout Partitioning on Pre-Configured ANN Configurations

Dataset Partitioning	Training Accuracy	Validation Accuracy	Testing Accuracy
70-15-15	0.82	0.81	0.72
80-15-5	0.79	0.73	0.75
90-0-10	0.77	N/A	0.71
80-10-10	0.87	0.81	0.89

2.3.3 K-fold Validation:

In searching for ANN hyperparameters and parameters, we employed K-fold validation to get very good estimation of the expected performance of each candidate ANN. The value $K=10$ was chosen, not only because this is the general practice in many other studies, but also because in each of the K iterations, the dataset can be partitioned into 80-10-10 for training-validation-testing which is consistent to what was earlier observed to be a good partitioning of the dataset.

Furthermore, recall that the dataset consists of 15 readings for each of the 100 different material samples (10 different material samples for each of the 10 odor classes). Data stratification was done in distributing the 10 different samples per odor class to the 10 folds, so that each fold contains samples from all the odor classes. However, while the material samples were distributed evenly, the 15 readings of each material sample were crucially grouped together under the same fold so that these similar readings from the same material in the test subset were not used to train the network, thus ensuring the integrity of test results.



Figure 7. E-nose Prototype

3 Results and Discussions

The e-nose prototype is shown in Figure VII. The hyper parameters were tuned via random search to configure an optimum network architecture. The best odor classification performance, in terms of precision, recall, and f-measure, was achieved using a 10-3-9-10 ANN shown in Fig. 8. The ReLU activation function was implemented for the hidden layers of the final ANN because of its fast learning capability while yielding good accuracy. Training was done on a min-max normalized dataset and 10-fold cross-fold validation was implemented to estimate the generalizing power of the different neural network configurations.

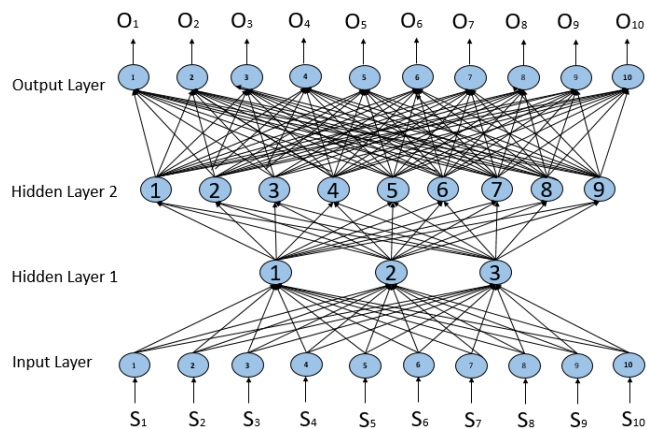


Figure 8. Final ANN Architecture

The consolidated confusion matrix in Table IV shows very accurate classification to at least half of the 10 classes. Accuracies of over 90% were obtained for the following 5 odor classes: Fragrant (96%), Fruity (98%), Chemical (99%), Peppermint (98%) and Popcorn (90%). In addition, the performance on Woody is within an acceptable value of 82.7%. A closer investigation of the sensor measurements per odor class reveals that the classes with relatively inferior results (decaying, pungent, citrus, sweet) have several sensors with comparatively close measurements. For example, about 30% of decaying samples were incorrectly classified as citrus, and the sensor readings for MQ3, MQ7, MQ8, MQ135, MQ138 were within very close ranges.

Table 4. Final ANN Confusion Matrix

Class	Fragrant	Woody	Fruity	Chemical	Minty	Decaying	Pungent	Citrus	Popcorn	Sweet
1	144	5	1	0	0	0	0	0	0	0
2	13	124	1	0	12	0	0	0	0	0
3	2	0	147	1	0	0	0	0	0	0
4	0	1	0	149	0	0	0	0	0	0
5	0	0	0	0	148	0	2	0	0	0
6	0	0	0	0	0	71	6	44	17	12
7	0	0	0	0	28	4	83	0	24	11
8	0	28	1	0	2	17	0	102	0	0
9	0	0	0	0	0	0	1	0	136	13
10	0	0	0	0	0	4	28	0	30	88

The MQ-series gas sensors have partial sensitivity and they can respond marginally to a range of gases. A boxplot of each sensor for each odor class in Fig. 9 reveals the overlaps between the ranges of measurements on different scents, making the classes harder to set apart. For this reason, more sophisticated sensors may be employed for future studies. Moreover, boxplots for Sensor 9 (MQ135) and Sensor 6 (MQ7) are predominantly similar. It seems that removing one of these sensors is not likely to impact the e-nose performance negatively.

Table 5. Class Precision, Recall and F-Measure for the different Odor Classes

Class	Precision	Recall	F-measure
Fragrant	0.960000000	0.905660377	0.932038835
Woody	0.826666667	0.784810127	0.805194805
Fruity	0.980000000	0.980000000	0.980000000
Chemical	0.993333333	0.993333333	0.993333333
Peppermint	0.986666667	0.778947368	0.870588235
Decaying	0.473333333	0.739583333	0.577235772
Pungent	0.553333333	0.691666667	0.614814815
Citrus	0.680000000	0.698630137	0.689189189
Popcorn	0.906666667	0.657004831	0.761904762
Sweet	0.586666667	0.709677419	0.642335766

Additionally, the confusion matrix in Table IV shows that classes with inferior results (decaying, pungent, citrus, sweet) have a good number of samples misclassified as belonging to one of the other classes. For instance, 9% false of the decaying samples were misclassified as citrus and 11% of the citrus samples were misclassified as decaying. Using a second pass two-odor ANN classifier chained to the original 10-class ANN network, in a future study, may help address some of these error-prone classifications.



Figure 9. Measurement Box Plots per Sensor

4 Conclusion

This study presented the feasibility of making an e-nose prototype intended for general odor classification. The system was realized by integration of hardware unit, which yields multi-dimensional odor measurement data for each sample, with an ANN that interprets and relates the data to a target class. The hardware unit is mainly comprised of an array of ten gas sensors, a microprocessor which serves as the data acquisition and interfacing component, a supply system for power requirements, a protection circuitry against voltage and current transients, an exhaust system to facilitate removal of gas and generate inward flow of air, and a compact chamber designed to allow easy sampling. The prototype served to provide a working dataset for ANN training purposes and to test the performance of the end design e-nose system. The final e-nose classifier with topology 10-3-9-10 and ReLU activation function returns very accurate predictions to at least half of the classes, namely fragrant (96%), fruity (98%), chemical (99%), peppermint (98%) and popcorn (90%).

Future studies may investigate improving the generalization power of the ANN by exploring other types of ANN architectures, utilizing more sophisticated hardware sensors, and even augmenting on the gathered data to allow for better ANN training.

5 Acknowledgment

The research is supported by the Department of Science and Technology - Engineering Research and Development for Technology, Philippines. The authors wish to express deep gratitude to Dr. Rosula Reyes, Dr. William Yu, Dr. John Paul Vergara and Dr. Ariel Maguyon for their valuable inputs, motivation and support in this study.

REFERENCES

- [1] Barwich, "A sense so rare: measuring olfactory experiences and making a case for a process perspective on sensory perception." *Biol Theory* 9:258–268, 2014.
- [2] L. Harman, "Human relationship with fragrance," In: *The chemistry of fragrances: from perfumer to consumer*. Royal Society of Chemistry, Cambridge, 2006, pp 1–2.
- [3] D. Schild and J.W. Gardner, "Detection and coding of chemical signals: a comparison between artificial and biological systems," University of Warwick, 1991.
- [4] N. Barsan, and U. Weimar, "Electronic nose: current status and future trends," Institute of Physical and Theoretical Chemistry, University of Tübingen, Germany, 2008.
- [5] K. Persaud, and G. Dodd, "Analysis of discrimination mechanisms in the mammalian olfactory system using a model nose," *Nature*. 299:352–355, 1982.
- [6] P.E. Keller, R.T. Kouzes, and L.J. Kangas, "Three neural network based sensor systems for environmental monitoring," *IEEE Electro 94 Conference Proceedings*, Boston, MA, 1994, pp.377-382.
- [7] R.J. Lauf and B.S. Hoffheins, "Analysis of liquid fuels using a gas sensor array," *Fuel*, vol. 70, 1991, pp. 935-940.
- [8] H.V. Shurmur, "The fifth sense: on the scent of the electronic nose," *IEEE Review*, March 1990, pp. 95-58.
- [9] Amico, A. Natale, C. Paolesse, and R. Macagnano, "Olfactory systems for medical applications" *Sens. Actuators B Chem.*, 1, 2008, 458–465.
- [10] I.A.Casalinuovo, D. di Pierro, M. Coletta, P. di Francesco, "Application electronic noses for disease diagnosis and food spoilage detection," *Sensors*, 6, 2008, 1428–1439.
- [11] H. Sun, F. Tian and Z. Liang, "Sensor array optimization of electronic nose for detection of bacteria in wound infection," *IEEE Transactions on Industrial Electronics*, Volume 64, Issue 9, 2017.
- [12] Norah Trent, "Global Electronic Nose Industry 2016 Market Research Report," 2016.

- [13] J. Castro, A. Ramanathan, and C. Chennubhotla, "Categorical dimensions of human odor descriptor space revealed by non-negative matrix factorization," 2013.
- [14] M. Roa and P. Fernandez, "A study of the effect of network architecture in artificial neural network performance applied to electronic olfactory device, National Graduate Student Leadership and Research Conference, Laguna, Philippines, 18-19 August 2016.
- [15] M. Roa and P. Fernandez, "Study on optimization of artificial neural network generalization power based on architecture," Proceedings of 90th The IIER International Conference, Dubai, UAE, 1st-2nd January 2017, ISBN: 978-93-86291-78-3.
- [16] M. Roa and P. Fernandez, "An empirical study of different artificial neural network activation functions applied to five classification problems," submitted, 2017.
- [17] M. Roa and P. Fernandez, "Development of an Electronic Nose for Olfactory System Modelling using Artificial Neural Network," 2018. [Online]. Available: <https://www.researchgate.net/project/Development-of-an-Electronic-Nose-for-Olfactory-System-Modelling-using-Artificial-Neural-Network>
- [18] S. Glen, "Alpha Level (Significance Level): What is it? Retrieved," 2012. from: <http://www.statisticshowto.com/what-is-an-alpha-level/>