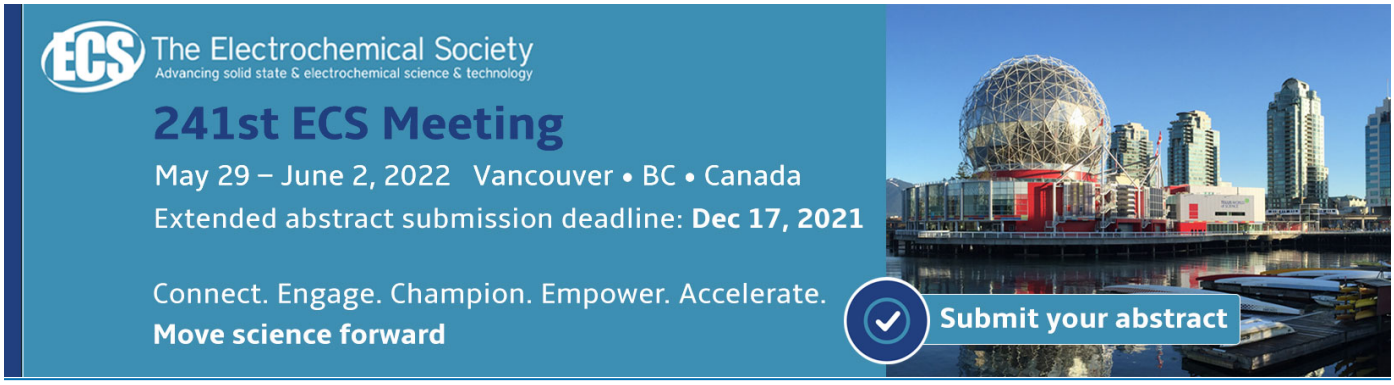
A Deep Belief Network and Dempster-Shafer Theory Multiclassifier for Reliability of Wind Turbine System



A Deep Belief Network and Dempster-Shafer Theory Multiclassifier for Reliability of Wind Turbine System

Bin Yu

Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, MA 01003-9292 USA

E-mail: biny@umass.edu

Abstract. Object. Reliability of wind turbine system could be made by predicting the status of power system in case it could remind engineers to fix in advance. There are multiple variables associate with system. An approach for classification according to these variables needed. Method. I proposed a deep belief network and Dempster-Shafer Theory (DBN-DS) by multiclassifier for whether wind turbine system needed to fix. DS would make a correct decision delete from DBN when model outputted incorrect result. And diagnose the wind turbine system to make a correct decision. Result. The new method was validated on wind turbine system data in 42-day intervals. The accuracy of SVM, KNN, MLP, DBN and DBN-DS were 95.1%, 96.1%, 91.2%, 97.0% and 97.5% respectively. Conclusion. This DBN-DS was more efficient that KNN, MLP, SVM and DBN. The performance was high because of binary Restricted Boltzmann Machines. The proposed method may be effective in decision support for diagnosing the wind turbine system.

1. Introduction

Wind turbine system is the most common facilities all over the world, with millions of cases constructed. As the renewal resources expanding, it is difficult for engineers to give the diagnose at the occasion of accident in the remote area.

Many scholars applied the DBS-DS in many subjects. *Duy Tang Hoang* applied DDB-DS model to bear fault diagnosis is a critical task which helps to reduce the cost of maintaining [1]. *Jae Kwon Kim* predict prostate cancer based on biopsy information [2]. The DBS-DS learns prostate-specific antigen (PSA), Gleason score, and clinical T stage variable information using three DBNs. Uncertainty regarding the predicted output was removed from the DBN and combined with information from DS to make a correct decision. *Kun Yu* proposed the framework utilizes a combined deep belief networks (DBNs) and Dempster–Shafer (D-S) theory fault diagnosis scheme and adopts a two-stage approach in classifying bearing fault conditions and fault severities [3].

For the part of wind turbine system reliability assessment, *C. Murray* proposed algorithms have been developed to enable the detection of faults and transient events in vibration signals [4]. The ability of the new algorithms to identify real-time features has been demonstrated using bearing seeded fault testing. *Lotfi Saidi* talked about a support vector regression (SVR) model was trained and tested for the prediction of the high-speed shaft bearing (HSSB) lifetime prognostics, showing the superiority of SK-derived indices of degradation assessment [5]. The experimental result verified the potential of the prognostics method using real measured data from a drivetrain wind turbine. *Surya Teja Kandukuri* reviewed the state-of-the-art in diagnostics and prognostics pertaining to two critical

failure prone components of wind turbines, namely, low-speed bearings and planetary gearboxes [6]. The survey evaluates those methods that are applicable to wind turbine farm-level health management and compares these methods on criteria such as reliability, accuracy, and implementation aspects.

Deep belief networks are an approach of deep learning and an efficient way to classify [7]. the Dempster-Shafer theory is a probability approach to improve the belief in Bayes problems [8].

In this paper, I proposed a DBN-DS multiclassifier for wind turbine system classified of reliability. The proposed DBN-DS adopted wind speed, grid power, rotor speed and least 7 features. When labels were generated by similarity measurement and clustering the historical data by Gaussian Mixed Model, the model was classified to output final multiclassifier result and assess the reliability of wind turbine system.

2. Materials and methods

This section I introduced the dataset, deep belief network and basic principle of Dempster-Shafer.

2.1. Data set

The data comprised 55-day wind turbine system from February 21, 2016 to April 16, 2016, and the time interval was 5 seconds, extracted from the electricity company. The twelve-input variables consisted of day internal, time internal, windspeed, grid power, rotor speed, PcsMeasuredGenSpeed, NacelleCabTemp, NacelleAirTemp, MainBearingGbSideTemp, GearboxOil, SumpTemp, GbLubOilFilterInletPre, Mean PitchPosition.

2.2. Gaussian mixture model

Gaussian Mixture Model (GMM) was an approach to analysis nonparametric problems. *Charles E. Antoniak* proposed the Mixture of Dirichlet Process to apply in bioassay, discrimination, and regression [9]

A gaussian mixture model was a weighted sum of M component Gaussian densities as given by the equation [10].

$$p(x \mid \lambda) = \sum_{i=1}^{M} w_i g(x \mid \mu_i, \sum i) \tag{1}$$

Where x was D-dimensional continuous-valued data vector, w_i , i=1…, M, were the mixture weights, and $g\left(x \mid \mu_i, \sum i\right)$, i=1…, M, were the component gaussian densities. Each component density was a D-variate Gaussian function of the form.

$$g(x \mid \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2} \mid \Sigma_i \mid^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_i)^{'}\Sigma_i^{-1}(x-\mu_i)\right\} \tag{2}$$

With mean vector μ_i and covariance matrix $\sum i$, The mixture weights satisfied the constraint the $\sum_{i=1}^{M} w_i = 1.$

The complete gaussian mixture model was parameterized by the mean vectors, covariance matrices and mixture weights from all component densities. There parameters were collectively represented by the notation.

2.3. Deep belief network

A deep belief network was deep neural network based on backpropagation network, which was BP-DNN. Obviously, the hidden layers were large, at last 2 layers. Then the value of weight which was almost 0 was propagated to the lowest hidden layer. Until 2006, *Hinton* proposed a layer-by-layer greedy method for pre-training restricted Boltzmann machines, which greatly improved the efficiency of training and improved the problem of local optimization very well [11].

As the Figure 1 showed, I illustrated the structure of DBN based on the 3 hidden layers network. The network was made up of 3 Restricted Boltzmann Machine (RBM) units. The RBM had two layers, the upper layer is the hidden layer and the lower layer is the visible layer. When stacked into DNN, the output layer (hidden layer) of the previous RBM was used as the input layer (visible layer) of the next RBM unit, which is stacked in turn to form the basic DBN structure. Finally, an additional output layer was added, which was the final DBN-DNN structure.

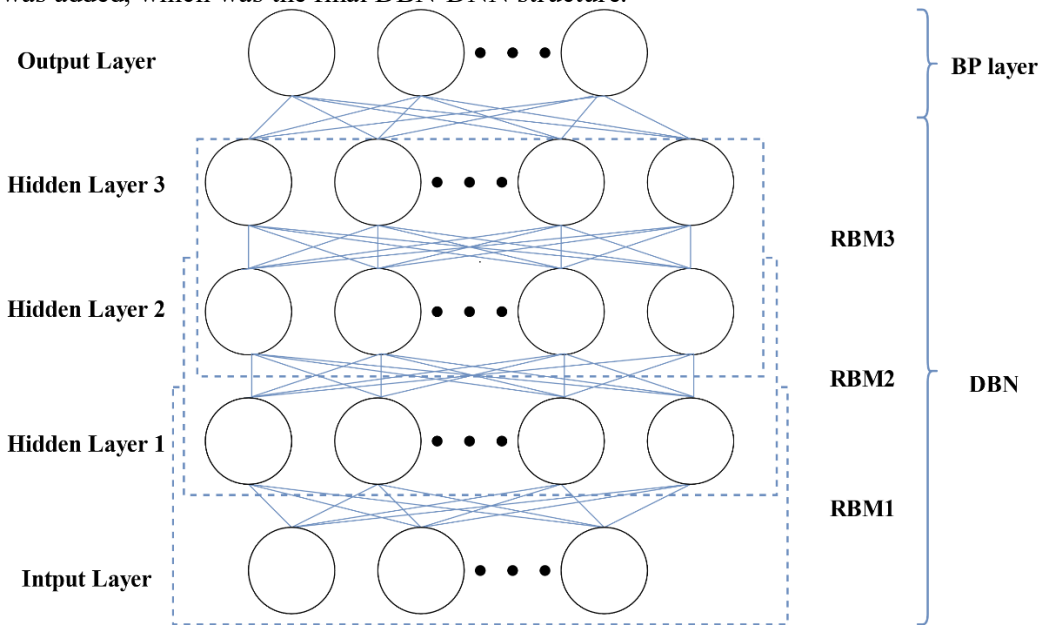


Figure 1. Structure of Deep Belief Network.

In this study, I constructed a classifier for ten-input and three-output variables to construct a multiclassifier, as showed in Figure 2. I created each classifier for each variable. I was to construct the multiclassifier to classify each variable. The proposal of paper intended to make the linear model to get the classified result combined with DS theory.

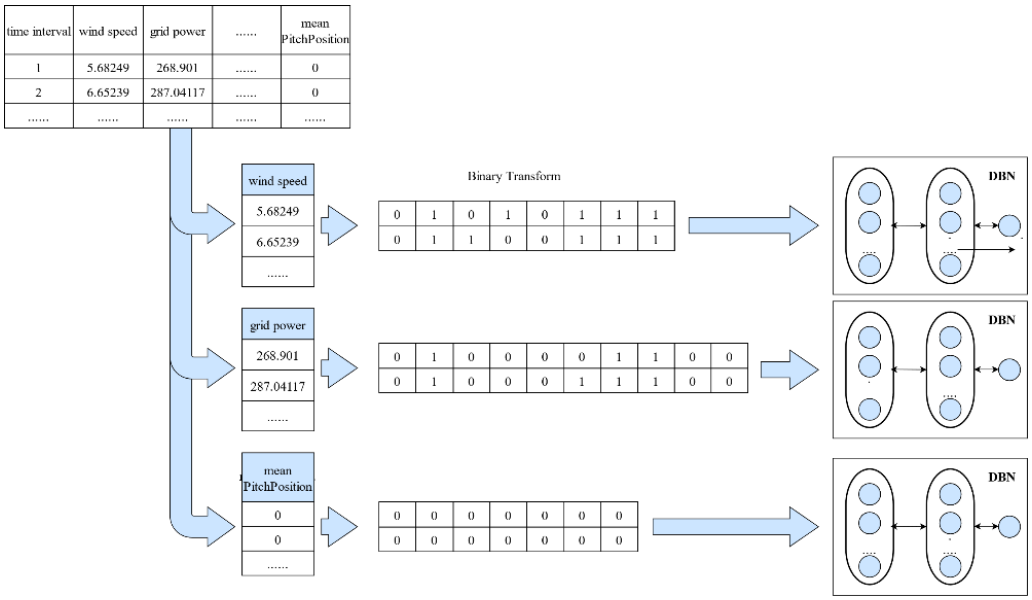


Figure 2. Multi DBN Classifiers.

2.4. Dempster-shafer information fusion

Dempster-Shafer was a mathematical theory, which was proposed by *G Shafer* [12]. It was an efficient approach in the decision field. I added this theory to output label result, which improved the model performance. I illustrated the way to implement in this section.

If X was the global variable: the set was all possible states of system under all options. The power set was 2^X , including the empty set ϕ . For example, when $X=\{a,b\}$ then in (3)

$$2^X=\{\phi,\{a\},\{b\},\{X\}\} \tag{3}$$

The element of the power set delegated the concerning the state of system. Including all and only the state when the proposition was true.

The evidence assumed a belief mass to every element in the power set. $m:2^X\to[0,1]$ was named a basic belief assignment (BBA). When the power set was empty, it could be assigned zero. In addition, the masses of all the members of power set were one.

The mass m(A) of A, a section of the power set, illustrated the proportion of evidence which supported the actual state belongs to A but to no subset of A. The value of m(A) maintained only to the set A and made no additional claims about any subsets of A.

From the mass definition, the upper and lower bounds of a probability interval could be defined. This interval contained the precise probability of a set of interest (in the classical sense), and was bounded by two non-additive continuous measures called belief (or support) and plausibility:

$$bel(A)\le P(A)\le pl(A) \tag{4}$$

The belief bel(A) with a set A was assigned as the sum of all the masses of all subsets:

$$bel(A)=\sum_{B|B\subset A}m(B) \tag{5}$$

The plausibility pl(A) was sum of all the masses of sets B connecting sets A.

$$pl(A)=\sum_{B|B\cap A\neq\phi}m(B) \tag{6}$$

The two measures showed in (7).

$$pl(A)=1-bel(\bar{A}) \tag{7}$$

And finally, for limited A, assigned the belief measure bel(B) for all B of A, then calculate mass m(A) with the inverse function.

$$m(A)=\sum_{B|B\subset A}-1^{|A-B|}*bel(B) \tag{8}$$

In equation (7), |A-B| was the different distance between sets A and sets B.

It came from (7) and (8) that, for a finite set X, one needed to know only one of the three (mass, belief, or plausibility) to deduce the other two [13].

The Dempster's rule of combination was the appropriate fusion operator. This principle was belief among sources. Cumulative fusion meant that all probability masses from the different were reflected in the belief, so mass was ignored without probability. Finally, the combination was work out from the two sets of m1 and m2:

$$m_{1,2}(\phi)=0 \tag{9}$$

$$m_{1,2}(A)=(m_1\oplus m_2)(A)=\frac{1}{1-K}\sum_{B|B\cap A\neq\phi}m_1(B)m_2(C) \tag{10}$$

Where

$$K=\sum_{B\cap C\neq\phi}m_1(B)m_2(C) \tag{11}$$

In the equation (11), K was a measure of amount of conflict between two mass sets.

3. Result

3.1. Data description

The eigenvectors of wind turbine system were ten vectors showed in the section 2.1, excluding day internal and time internal. Among 954683 time-interval, mean and standard deviation of wind speed, the first eigenvector, was 7.19 and 5.24, respectively. And then these of grid power was 607.23 and 767.68, respectively. the rest of eight eigenvectors were not repeated to calculate.

In this experiment, original data existed nan and outlier data. To improve accuracy, I adopted to filter the outlier data by calculating the quartiles. Three Sigma Rule was a normal approach to deal with outlier data [14]. Therefore, showing in the Figure 3, I used quartile and three-sigma rule to calculate the upper and lower boundary of quartile and these of three-sigma rule with the wind speed.

The two methods were independent and compared the filtering results. The interquartile method was to calculate the upper and lower quartiles. Firstly, sort the data, and then the median subtracted the lower interquartile as the lower boundary, and the median plus the upper interquartile as the upper boundary. The three-sigma method was to calculate the mean and standard deviation of the data, and then the mean subtracted three times the standard deviation as the lower boundary, and the mean plus three times the standard deviation as the upper boundary. Identify as outlier beyond the upper and lower boundaries.

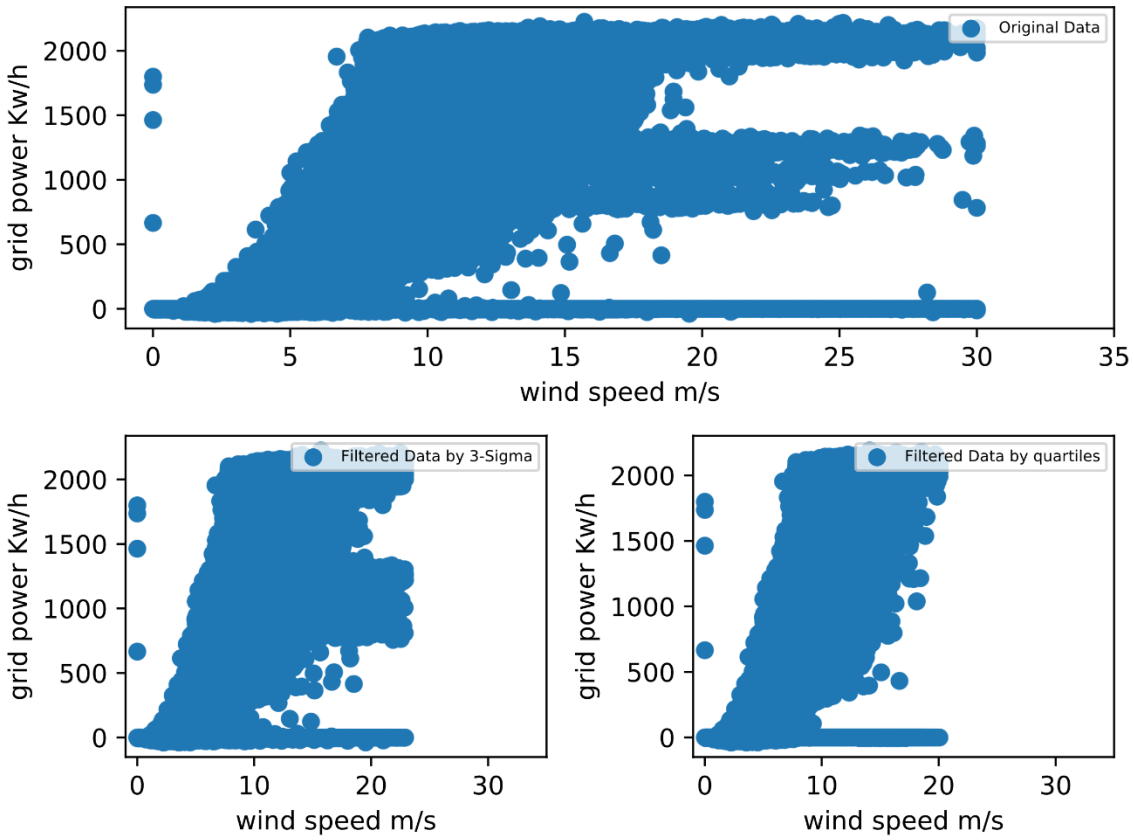


Figure 3. Comparison of outlier data processing.

Showing in the Figure 3, The filtering performance of interquartile was better than it of three-sigma.

3.2. GMM cluster and similarity measurement

According to data description, there were unlabeled. To label the status of wind turbine system such as health, risk and fix, the original data was clustered by GMM to output clustered result and classified by similarity measurement.

3.2.1 GMM cluster process.

In the original set, to seek to represent the linear combinations of a small number of eigenvectors and minimize the mean-squared reconstruction error. I adopted the Principal Component Analysis (PCA) to get principal components in this dataset. PCA was mostly used as a tool in exploratory data analysis and for making predictive models. It was often used to visualize genetic distance and relatedness between populations.

PCA was either done by singular value decomposition of a design matrix or by doing the following 2 steps: calculating the data covariance matrix of the original data, performing eigenvalue decomposition on the covariance matrix [15].

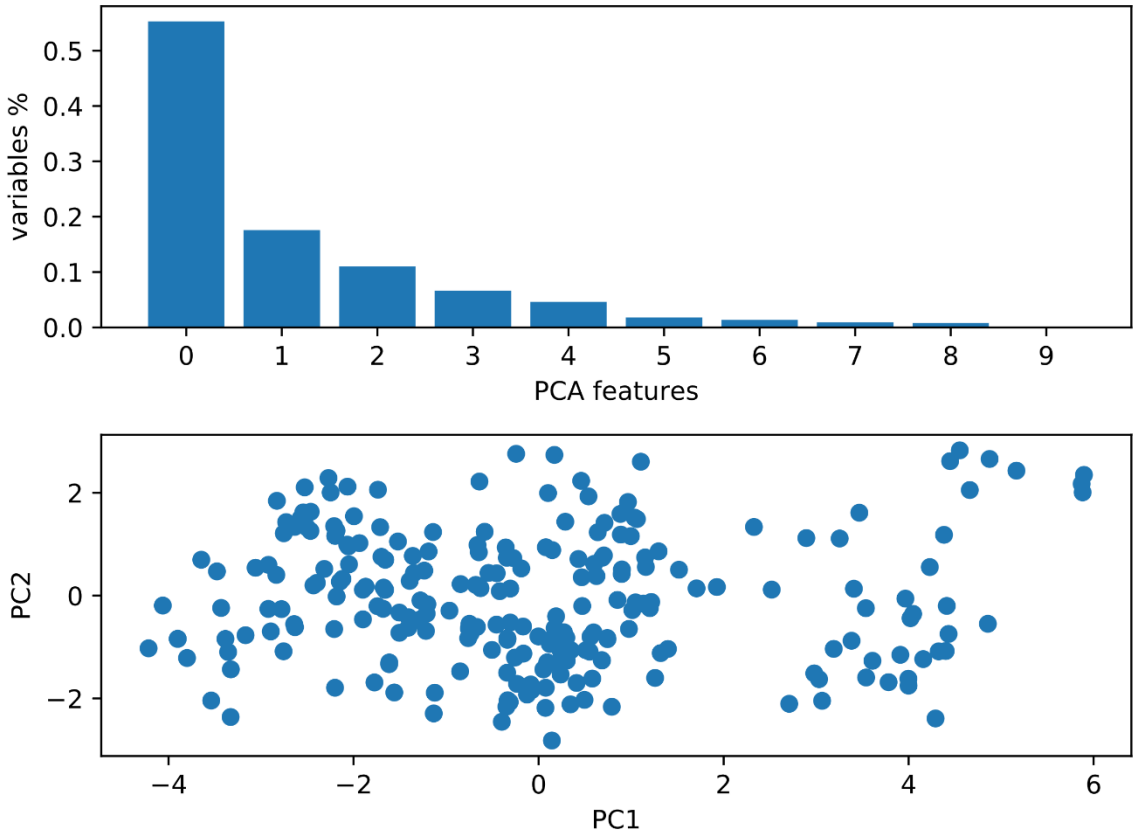


Figure 4. PCA Transforming Result.

Firstly, I applied the standard scale for the original data from 0 to 1 which was fitted to calculate the explained variance ratio shown in the Figure 4. The upper graph was the explained variance ratio. There were 3 groups in this histogram. PCA feature 0 was group 1, PCA feature 1,2,3,4 was group 2 and PCA 5,6,7,8 was group 3. In the meanwhile, I chose the 0.4% data to visual the scatter and in the lower graph got at least three distinguishable clusters. This factoid told that the observations in the dataset could be grouped. I did this to notice if there were any clear clusters.

In this step, I used GMM clustering to input entire principal components. To do this, I first fitted these principal components to the GMM and determined the best number of clusters. Determining the number of clusters for our GMM could be done by measuring the lower bound value on the log-likelihood. Much like the scatter plot in Figure 4 for PCA, GMM scatter plot indicated the percentage of variance explained, but in slightly different terms, as a function of the number of clusters.

Figure 5 showed that after clustering when number of components was 3, which was an elbow point, at the changing in the value of the lower bound was no longer significant. Therefore, I could get k equaling 3 and proceed to the next step in the process.

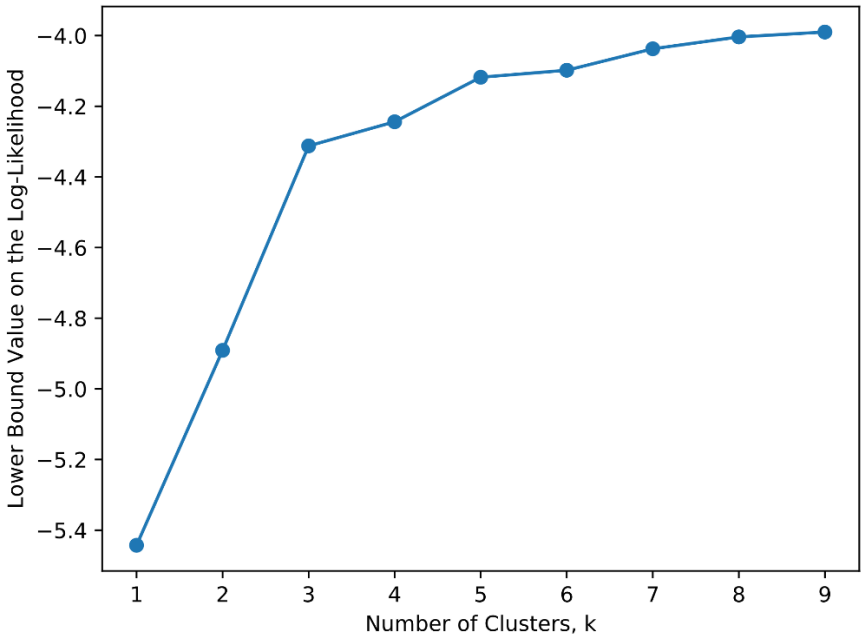


Figure 5. Lower Bound Value of K Cluster.

As the number of clustering center was three, GMM fitted the original data and visual the clusters in Figure 6. And I could get there were three conditions in the wind turbine system. The percentage of condition 1, condition 2 and condition 3 were 16.34%, 58.09% and 25.57% respectively.

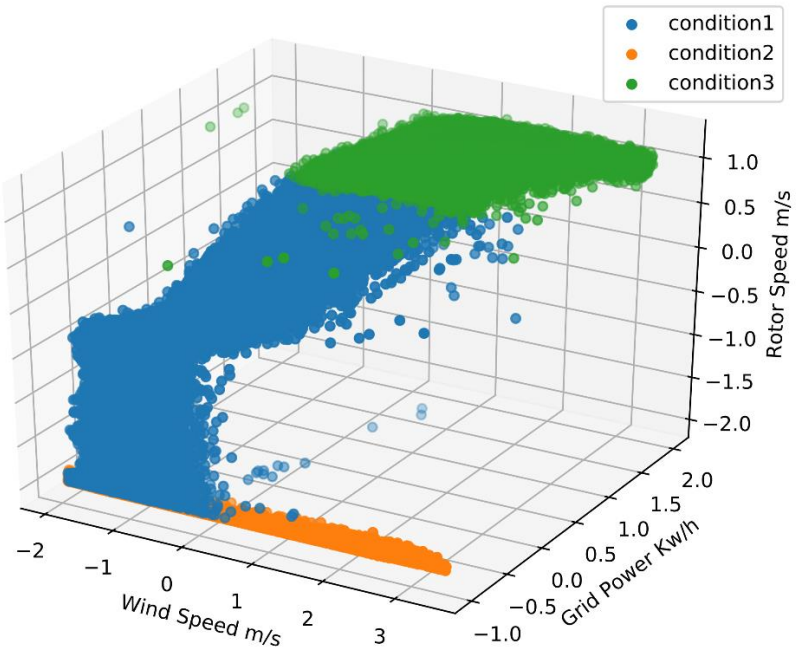


Figure 6. GMM Cluster.

Specifically, the number in three conditions showed in the Table 1. The percentage of condition 2 was the largest among three groups.

Table 1. Summary of GMM Cluster.

Original Set (n=675230)		
Condition 1	Condition 2	Condition 3
346884	211456	116890

3.2.2 Similarity measurement.

Calculate the similarity measurement which was ϕ showing in the equation (10). And in the PCA the first principal variable was ϕ_1, the second principal variable was ϕ_2, the third principal variable was ϕ_3.

$$\phi=0.5809*\phi1+0.2557*\phi2+0.1634*\phi3 \quad (10)$$

In addition, I used the time slide to resize the ϕ. ϕ reflected the dynamic process of the degradation, the larger the value, the more obvious the degradation. Figure 7 showed segment boundary was mean of ϕ and mean of ϕ pulsing variance of ϕ. It meant that ϕ was less than mean boundary, which was healthy status and mean + variance boundary which was risk status by reliability analysis. In this step, I labeled the original data by similarity measure, which was classified as the health, risk, and fix.

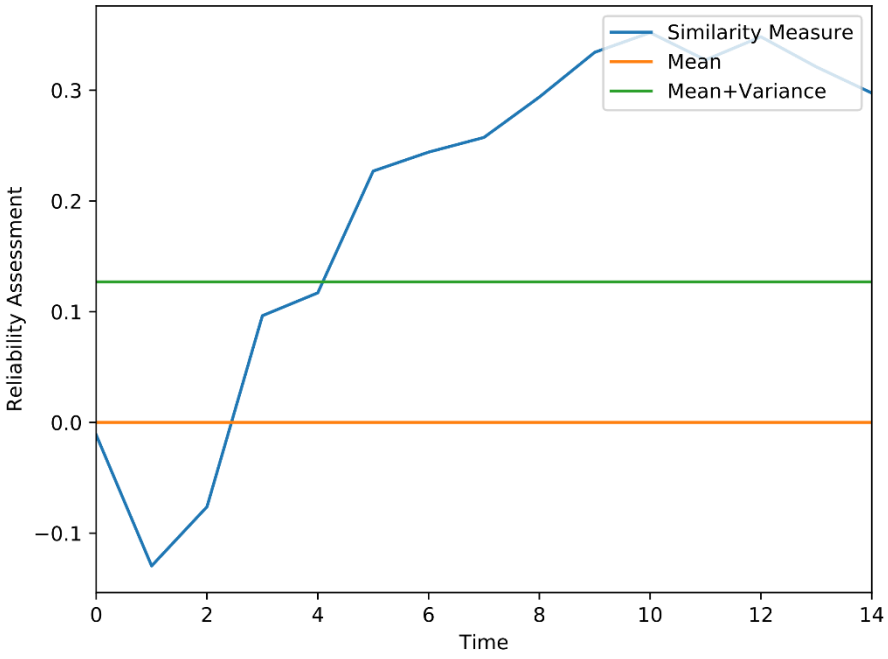


Figure 7. Similarity Measurement.

I split the 67% dataset as the train data, and the rest was treated as validation section. There were 675230-time interval in original data in case reduce the size of model. I reduced the size of dataset and randomly choice 1% in entire data. In the training section, the mean and standard deviation of wind speed were 0.01 ,1.00, respectively. In the validation section, they are -0.02,0.99. the difference in values between the training and validation sets was not large. Although the mean of wind speed in validation section was negative, it was irrelevant with accuracy of model.

In the Table 2. The number of risk status was larger than that of health in validation set as same as training set. That why I proposed this model to do reliability analysis.

Table 2. Summary of Training and Validation Set.

Training Set			Validation Set		
(n=4523)			(n=2229)		
health	risk	fix	health	risk	fix
1476	2268	779	670	1184	375

3.3. DBN-DS multiclassifier

The main structure of DBN and DS theory showed in the Figure 8. The training set was first changed to binary form. The initial wind speed value was assigned as 8 bits based on the maximal value in this column. As same as the binary format of wind speed, the initial grid power was set as 10 bits. Each binary data of feature was learned by the DBN classifier. Mentioned that there were different binary bits after original dataset was transformed. Basically, I set all input data was 32 bits, specifically, the thirty-second bit was sign bit.

Input data to DBN first layers which was 32 nodes, and output layers consisted of 3 nodes so that health, risk, and fix could be calculated with probability by SoftMax function. The DBN consisted of three RBM layers, with the number of nodes of each RBM the same as the number of input node. Finally, I calculated the probability of the put variable as DS and determined the final number of health, risk, and fix as the final outputs.

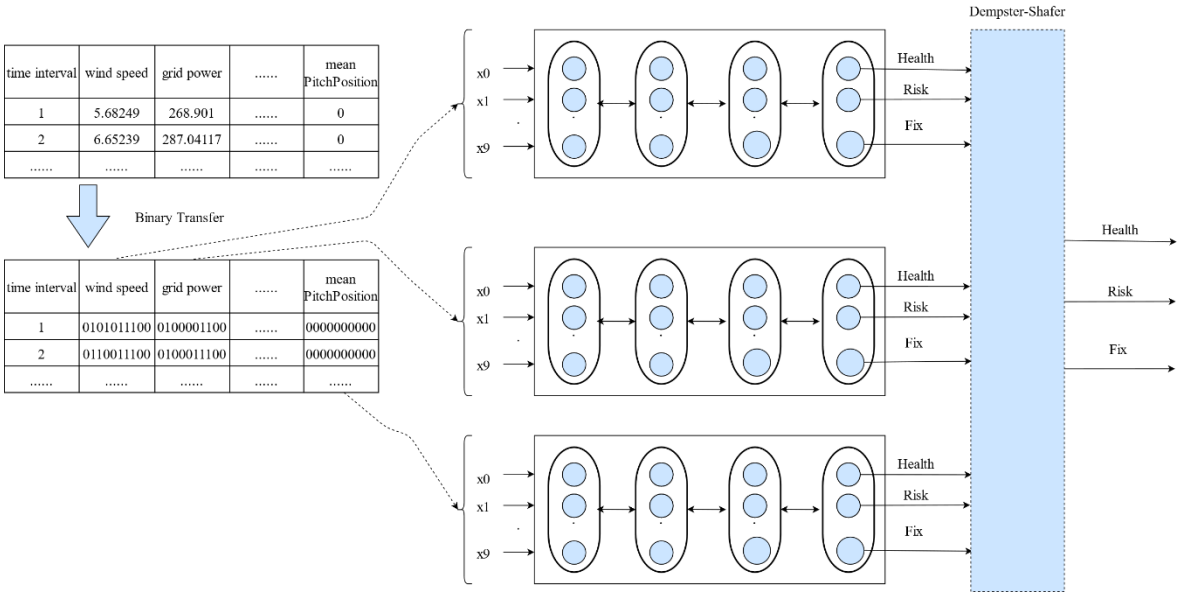


Figure 8. DBN-DS Multiclassifier.

This paper implemented the DBN-DS model by the NumPy and Scikit-learn package, running the value of loss showed in Figure 9.

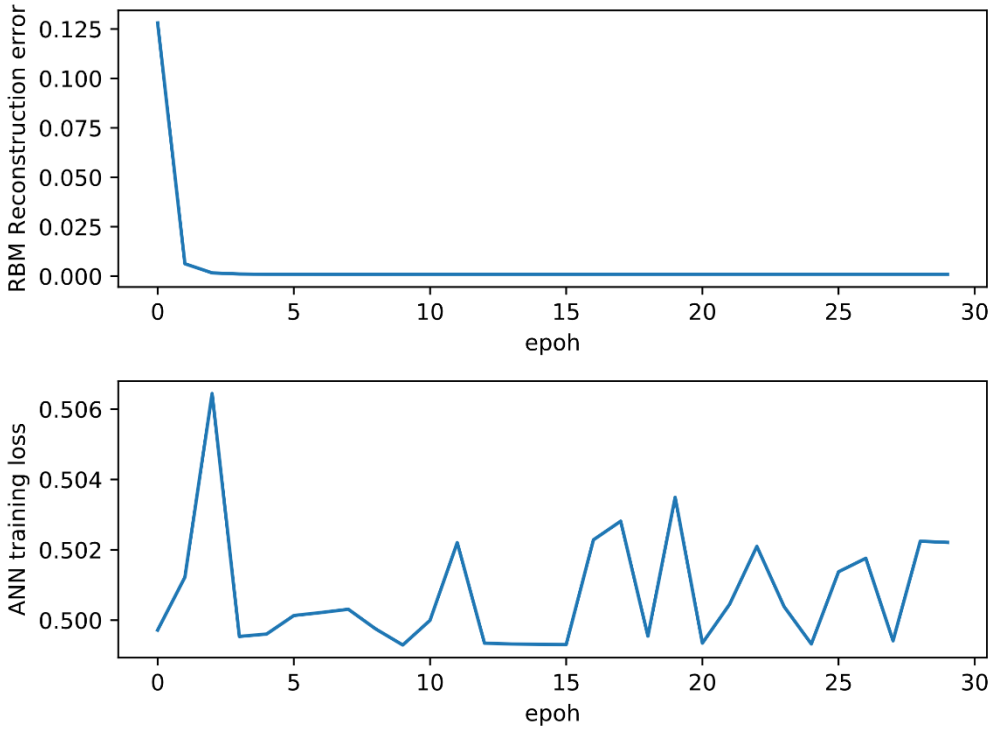


Figure 9. ANN Training Loss and RBM Reconstruction Loss.

3.4. Experiment

The process of training DBN-DS showed in Figure 5. To evaluate the DBN-DS model, the entire data was divided into 67% training set and 33% validation set. The control group included the support vector machine (SVM), Multilayer Perceptron (MLP), DBN and K Neighbors (KNN). The experiments compared the accuracy in Table 3.

Table 3. Summary of Validation Accuracy.

SVM	KNN	MLP	DBN	DBN-DS
0.951	0.960	0.912	0.970	0.975

4. Discussion and conclusion

I proposed a DBN-DS-based multiclassifier approach to predict the status of wind turbine system. The proposed method provided a predictive model to improve accuracy through deep learning and information fusion based on the relationship between data measured using company source data. The inputs included wind speed, grid power, rotor speed and soon. The output can be health, risk or fix. This approach was evaluated using an existing validated original dataset that included 2229-time interval.

The performance of the proposed DBN-DS was compared with that of the MLP, SVM, and RNN. The results showed that the proposed DBN-DS had better accuracy than all other methods.

Now, the proposed DBN-DS method was implemented as a research tool. Once the evaluation would be completed, the tool would be applying into easy-to-use decision support for engineer in electrical company.

References

[1] Hoang D T and Kang H J 2018 Deep belief network and dempster-shafer evidence theory for bearing fault diagnosis. *In 2018 IEEE 27th International Symposium on Industrial Electronics (ISIE)* 841-6
[2] Nakamura S, Senoh M, Nagahama S, Iwase N, Yamada T, Matsushita T, Kiyoku H and Sugimoto Y 1996 *Japan. J. Appl. Phys.* **35** L74
[3] Yu K, Lin T R and Tan J 2020 A bearing fault and severity diagnostic technique using adaptive deep belief networks and Dempster–Shafer theory. *Structural Health Monitoring*, **19(1)**, 240-61
[4] Murray C, Asher M, Lieven N, Mulroy M, Ng C and Morrish P 2014 Wind turbine drivetrain health assessment using discrete wavelet transforms and an artificial neural network
[5] Saidi L, Ali J B, Bechhoefer E and Benbouzid M 2017 Wind turbine high-speed shaft bearings health prognosis through a spectral Kurtosis-derived indices and SVR. *Applied Acoustics.* **120** 1-8
[6] Kandukuri S T, Klausen A, Karimi H R and Robbersmyr K G. 2016 A review of diagnostics and prognostics of low-speed machinery towards wind turbine farm-level health management. *Renewable and Sustainable Energy Reviews*. **53** 697-708
[7] Guo Y, Liu Y, Oerlemans A, Lao S, Wu S and Lew M S 2016 Deep learning for visual understanding: A review. *Neurocomputing.* **187** 27-48.
[8] Yager R R 1987 On the Dempster-Shafer framework and new combination rules. *Information sciences*. **41(2)** 93-137
[9] Antoniak C E 1974 Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The annals of statistics*. 1152-74
[10] Reynolds D A 2009 Gaussian Mixture Models. *Encyclopedia of biometrics*. 741
[11] Hinton G E, Osindero S and Teh Y W 2006 A fast learning algorithm for deep belief nets. *Neural computation.* **18(7)** 1527-54
[12] Shafer G 1992 Dempster-shafer theory. *Encyclopedia of artificial intelligence.* **1** 330-1
[13] Halpern J Y 2017 *Reasoning about uncertainty*. (MIT press)
[14] Pukelsheim F 1994 The three sigma rule. *The American Statistician.* **48(2)** 88-91
[15] Wold S, Esbensen K and Geladi P 1987 Principal component analysis. *Chemometrics and intelligent laboratory systems*. **2(1-3)** 37-52