Using Deep Learning to Predict User Rating on Imbalance Classification Data

Hendry and Rung-Ching Chen*

***Abstract*—Deep learning has demonstrated some remarkable successes. Deep learning can be characterized in several different ways, but the most important is that deep learning can learn higher-order interactions among features using a cascade of many layers. Despite its successes, it faces major challenges in computational complexity when using hyper-parameters and time-consuming processes to calculate many fully connected layers. We propose an alternate approach to facilitate deep learning with imbalanced datasets. We call our model Imbalance Deep Belief Network (IDBN). We incorporate ensemble models to enable the system to learn all the class targets (user rating) with a single deep learning base classifier model based on user comments. By this way, it can learn every rating with its fitting model. The main idea for the ensemble model is to give an alternate solution for base classifiers when selecting the best results. In the ensemble model, we use different features selection for the input layer. Using all features in the same model may increase computational complexity and consume much time. Further, some features can be assessed for significance. Insignificant features could be pruned in the classifier or be substituted with other features which are extracted from the sampling data. In the output layer, we apply voting methods to select user ratings as outputs from each base classifier to generate user rating prediction results with more extensive results. IDBN has more sustainably predicts bad and very bad' user ratings in imbalanced datasets than base models.**

***Index Terms*—Deep Learning, User Rating Prediction, Imbalance Classification.**

I. Introduction

In recent years, social media data has grown enormously, making social media an increasingly important research field. Researchers have proposed many techniques for using the information in social media including data mining, social network analysis, and recommendation systems.

With the success of deep learning approaches, machine learning has received greater attention from researchers [1], [2]. Deep learning has demonstrated remarkable successes through its capacity to create detailed (deep) models of complex multivariate in structured data. Deep learning can be characterized in several different ways, but the most important is that it is able to learn higher-order interactions among features using a cascade of many layers [3], [4], [5]. Despite successful breakthroughs in deep learning, it faces the great challenges in computational complexity with hyper-parameters and time-consuming processes when calculating many fully connected layers [6].

Many methods to address these challenges have been proposed, including Graphical Processing Unit (GPU) and hybrid methods. Although these methods have improved the speed of execution, complex computation faces a great challenge when incremental learning is necessary. Dropout is one alternative that speeds computation and reduces complexity by pruning nodes in the hidden layers. Although dropout can reduce the complexity of computation, dropout has problems handling imbalanced datasets. When deep learning trains the model of the dataset which consists largely of good impressions (for example user ratings with values 4 and 5), nodes with a bad impression (ratings 1 and 2) would likely result in very small values. When dropout detects these nodes with the lower impression than its threshold, they will be pruned. The model would likely fit and obtain good prediction results for ratings, yet handling of bad impressions is poorly executed.

Many authors have proposed classification processes for imbalanced datasets. The basic classifier methods such as Support Vector Machine (SVM)[7], [8], [9], [10] and Multi-Layer Perceptron (MLP)[11], [12] address classification problems. These algorithms are well known. Deep learning is a methodology applied to neural networks across many fields. Convolution Neural Networks [1], [2], Deep Belief Networks [3], [4], [5] and many other structures have been proposed to enhance the deep structure network ability. Since deep learning gains more attention in machine learning research fields for processing imbalanced dataset classification problems [8], we implement a Deep Belief Network (DBN) as the comparison method. We propose an alternative approach for learning of deep learning models in imbalanced datasets. We incorporate ensemble models for learning each class target (user rating) with one deep learning base classifier model based on user comments. By this way, we learn every rating with its fitting model.

Ensemble models offer a different way to learn where the main idea is to combine a set of models (base classifiers) to obtain more accurate and reliable results than a single model can obtain [13]. As explained in [14], weighted combinations of feature sets can be quite effective in classification problems. The main idea for the ensemble model is to give an alternate solution for base classifiers for selecting the best results.

In the ensemble model, we use different features selection for the input layer. Using all features in the same model can create computational complexity and make the running process time-consuming. Further, the significance of features can be assessed. The Less significant feature can be pruned by the classifier or substituted by other features extracted from the sampling data. This classification is based on two dimensions: how predictions are combined (rule-based and Meta-learning) and how the learning process is performed (parallel or sequential). In the rule-based approach, predictions are combined using a rule to average the performance.

Manuscript received July 26, 2018; revised December 27, 2018. This paper is supported by the Ministry of Science and Technology, Taiwan, with project number: MOST-106-2221-E-324-025 and MOST-106-2218-E-324).

Hendry is with Department of Information Management, Chaoyang University of Technology, Taichung, Taiwan, e-mail: hendry@uksw.edu

*Rung-Ching Chen is with the Department of Information Management, Chaoyang University of Technology, Taichung, Taiwan, e-mail: crching@cyut.edu.tw

Meta-learning techniques use predictions from component classifiers as features for a Meta-learning model. In the output layer, we apply voting methods to select the user rating as the output from each base classifier to generate user rating prediction results with more extensive results.

In this paper, we propose a combination of deep learning methods and ensemble models with imbalanced dataset. We treat each class of user rating as a base classifier, where several rating classifiers trained with different features are combined. In order to study the complements of our proposed method, three datasets are used in the experiments: a YELP dataset, an Amazon dataset, and aTrip-advisor dataset. In the experiments section, we also compare IDBN with the traditional classification algorithm LibSVM, basic neural network algorithm MLP, and basic deep learning algorithm DBN.

The main contributions of this paper are as follows. (1) The system combines deep learning and ensemble model to predict rating based on user comments from imbalance dataset. The method classifies learning model into the majority and minority classes. The classifier training avoids hidden layer tolerate to majority class which can suffer the minority class getting high accuracy value. (2) We propose an Imbalance Deep Belief Network (IDBN) to handle imbalance dataset to avoid modification of dataset. Some approaches using manipulation of minority class exist the bias of the results, and also faces over-fitting problems. Using IDBN, we do not manipulate the dataset. (3) The user rating is directly from user comments to incorporate text to vector algorithm and sentiment classification. The system learns the impression from the user's comments automatically and classifies it into good or bad impression vector using sentiment classification. By this way, the system automatically removes misclassified good rating from bad impression comments and vice versa.

The remainder of this paper is organized as follows. Section II is the related work. Section III is the deep learning algorithm that predicts user ratings in an imbalanced dataset. Section IV is the discussion of the results of the experiments and the performance of the deep learning training model. Finally, we give conclusions and ideas for future work in Section V.

II. Related Work

A. Classification in Imbalance Dataset

In machine learning research area, imbalanced datasets are a common problem. Imbalanced means that data points which represent different classes occur in different numbers [15]. In many types of research, if a minority class must be identified as the class which has the highest interest from a learning model, this will imply a great cost when it is not well classified [16]. To determine whether the dataset is imbalanced is not easy, because no specific threshold value is defined. Moreover, setting the threshold is also a challenge. Many authors have explored these issues [17], [18], [19]. They find that imbalanced datasets occur when a minority class is misclassified. Classification methods for imbalanced data can be categorized into three major groups:

1) Data Sampling: In this method, the dataset is modified in some manner to produce a loose order to make a balanced class distribution which allows classifiers to perform similar training models for each class label in the classification process. [20], [21], [22], [23] proposed this type of methodology to solve classification in imbalanced datasets.
2) Algorithmic Modification: In this method, the researchers modified their algorithm to adapt their base learning methods to be more attuned to the imbalanced class. This approach gives attention to minority classes in the training steps of the algorithm [24]. In machine learning research, this methodology has received more attention. When data sampling, we may miss important information in the data left behind. Conversely, modifying our dataset with sampling, boosting, or simulating may change the important information that we could collect from the data.
3) Cost-sensitive learning: The method is a combination of data sampling and algorithmic modification. This methodology incorporates solutions in the data level and algorithmic level to solve classifications problem in imbalanced datasets [25], [26].

In this paper, we proposed to predict user rating from imbalance dataset of user comments through algorithmic modification level. We incorporate deep learning and ensemble methods to let every class label (user rating) be trained specifically for each class. Then, we combine every base classifier to predict its user rating through the Bayesian optimal classifier.

B. Sentiment Classification

Sentiment Analysis is artificial intelligence research fields that focuses on user's opinions, behavior, and emotions through text mining [27]. Many fields of research have used sentiment analysis, including stock markets [28], [29], news articles [30], politic and public actions [31]. According to [32], sentiment analysis is a classification process. To perform this analysis, emotions from the text are treated as the factor under consideration. The emotions of the user can be obtained from the user's comments. Users' comments usually have an impression that is reflected in the user rating. In other words, user comments will have a good impression when they give a good rating score. Emotions can be measured using several factors such as subjective ratings and objective measurements [33]. Subjective ratings of emotions are typically collected through questionnaires. These generally consist of standardized labels or pictograms to represent emotions. In [34], emotions are used in subjective ratings using Self-Assessment Manikin (SAM) methods. SAM enables the participant to state their emotion which denoted using PAD (Pleasure, Arousal, and Dominance). Pleasure indicates how pleasant emotion is, arousal indicates how intense the emotion is, while dominance indicates how dominating it is. Objective measurements apply physiological sensors to assess the emotions of the participant.

Figure 1 shows a diagram cluster of emotions. Each quadrant contains several types of emotional factors. The emotions are divided into four groups, "Very Active," "Very Positive," "Very Passive," and "Very Negative" [35]. Each quadrant contains several types of emotional factors. To simplify the emotional factors of user ratings, we only

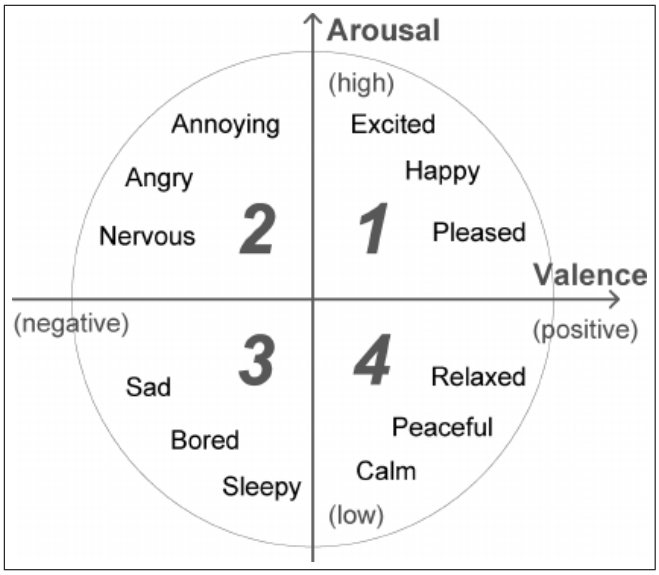

Fig. 1. The Evaluation Space of Emotion [35]

classify the emotional factors into eight emotion points using two from each quadrant as our emotion categories. These emotions factors are "Surprise," "Joy," "Anticipation," "Acceptance," "Sadness," "Disgust," "Anger," and "Fear." For our input vector, the sentiment is classified into two classes only, positive and negative impressions.

C. Deep Belief Network

Deep belief network (DBN) is a generative graphics model, which consists of multiple layers of hidden units, with connections between layers wherein each layer unit does not have any connections. DBN can construct its input from the original input node to choose the significant features to be used as the input layer in the training phase. These capabilities are based on the probabilistic reconstruction of their input, as the layers act as features detectors [36]. DBN may also be used to perform classification problems from its training model [4].

DBN consists of standard RBM units which have visible layer v_i, hidden layer h_j and a matrix of weights, $W = [w_{i,j}]_{mxn}$, where W is the connection v_i between h_j. RBM bias is represented by a_i for the visible layer and b_j for the hidden layer. The energy function for a configuration (v, h) is defined as equation (1).

$$E_{(v,h)} = -\sum_{ij} v_i W_{ij} h_j - \sum_i a_i v_i - \sum_j b_j h_j \qquad (1)$$

This energy function is a type of probability distribution over the input vector which is defined as $P(v)$ as shown in equation (2).

$$P(v) = \frac{1}{Z} \sum_h e^{-E_{(v,h)}} \qquad (2)$$

In equation (2), the variable z is a partition value over all possible configurations. The visible units of RBM can be multinomial, although the hidden units are Bernoulli. In this case, the logistic function for input units is replaced by the softmax function. Equation (3) shows the function.

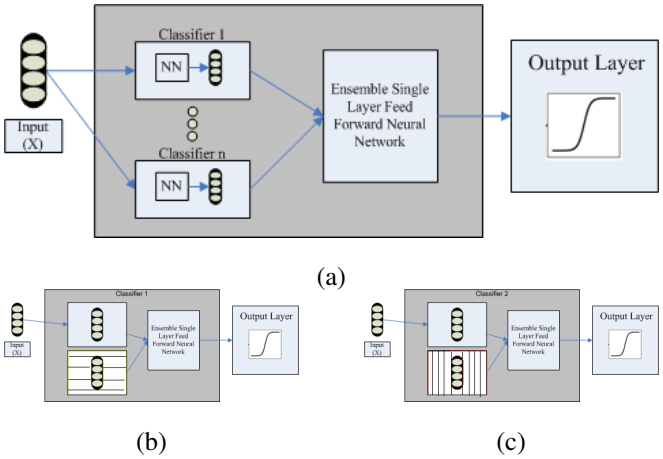

Fig. 2. The Ensemble Process. (a). The Ensemble Model with Different Base Classifiers, (b) and (c) Base Classifier with Different Features

$$P(v_k^i = 1 \mid h) = \frac{exp(a_i^k \sum_j W_{ij}^k h_j)}{\sum_{k'=1}^{K} exp(a_i^{k'} + \sum_j W_{ij}^{k'} h_j)} \qquad (3)$$

where K is a constant value, $\forall k \in K$.

D. Ensemble Model

The main idea for the ensemble model is to give an alternate solution for base classifiers to select the best results. In the ensemble model, we could use different features selection for the input layer. Using all features in the same model can generate computational complexity, lengthening processing time. Further, the significance of some features may be assessed. Less significant features can be pruned in the classifier or could be substituted by other features extracted from the sampling data.

Figure 2 shows the ensemble process for each different feature. The base models (b) and (c) have different features which are combined with the input node. The training process will take place in each classifier. This classification is based on two dimensions: how predictions are combined (rule-based and meta-learning) and how the learning process is done (parallel or sequential) [13]. In the rule-based approach, predictions will be combined using the rule to average the performance. Figure 2 (a) combines the base models (b) and (c) to generate user rating prediction results.

III. Propose Methodology

In this section, we present an ensemble model incorporating deep learning in an imbalanced dataset environment to predict user ratings. Figure 3 shows the proposed system architecture.

The system begins by collecting user comments from the product reviews. Incorporating Sentiment Analysis, we construct our input vector. The system opinion lexicon is a feature vector which we extract from the user comments database. The system constructs the feature vector $Ve = [Ve_(u, bu)]_{k,l}$ as 50 dimensions of opinion lexicon based on the polarity of the words. where u is set of users, v is set of business, k is total user review, and l is a total business review. The feature vector consists of 25 dimensions of good impression and 25 dimensions of the bad impression, as shown in Figure 4

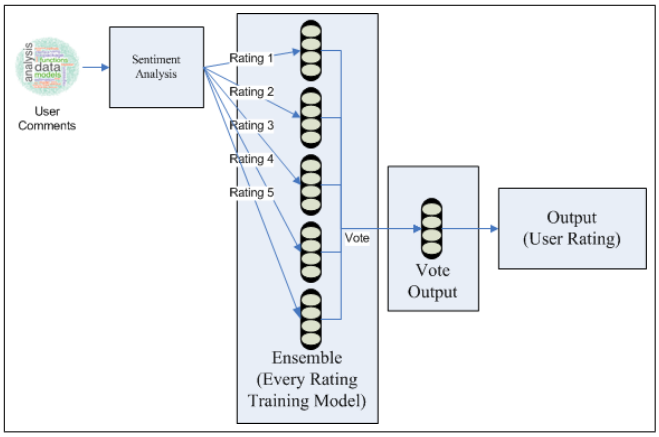

Fig. 3. The System Architecture

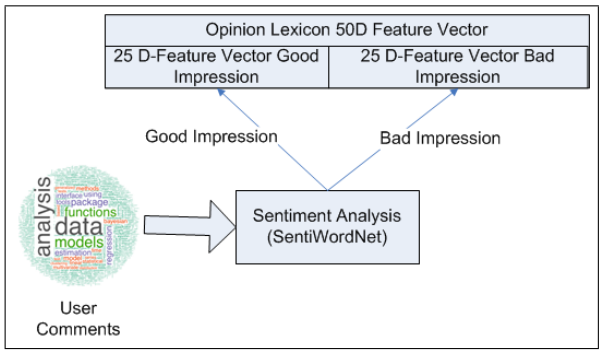

Fig. 4. The Sentiword Generating Feature Vector

Ensemble models are used to learn each rating with a deep learning training model. Figure 5 shows the baseline deep learning model for learning user ratings. For each baseline, the training models training phase has two main parts. The first phase constructs the input layer with the features extraction model. We use a Restricted Boltzmann Machine (RBM) to learn the significant features from the input nodes. This phase consists of two steps:

1) Each vector ve from the set of vectors Ve it will be used as input v_i for the RBM learning process. Hidden units h_j can be derived from

$$p(h_j = 1|v) = \sigma(\sum_i w_{ij} v_i + b_j) \tag{4}$$

Where $\sigma(.)$ represents the sigmoid function and b_j is the bias of the hidden layer.

2) Reconstruct each visible vector as v with the following equation which is similar to equation (4)

$$p(v_i'|h) = \sigma(\sum_j w_{ij} h_j + a_i) \tag{5}$$

Where $\sigma(.)$ represents the sigmoid function and a_i is the bias for the reconstructed new vector layer. a_i is the bias that occurs when the input layer is reconstructed into a new input vector layer from RBM learning.

The second phase is to learn the user rating from the feature map (X). We use a back-propagation neural network to calculate the output layer (Y). Equation (6) shows the computation of the weighted sum net_h. Theta $(\theta_h\ )$ is the bias for each hidden node and W_{ih} represents the weight of the connection between input node Xi and hidden node h.

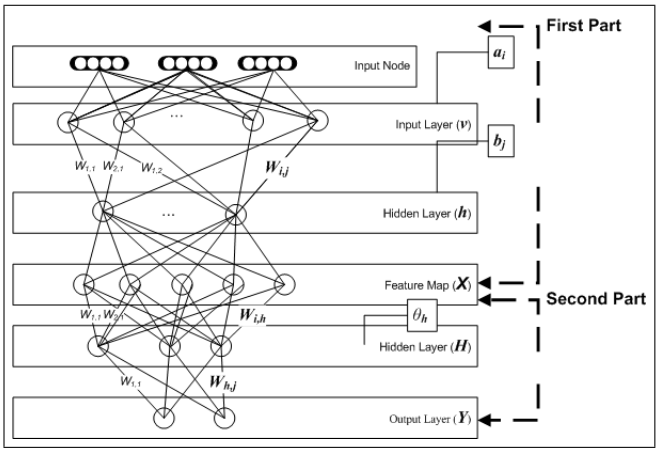

Fig. 5. Base Classifier Training Model

$$net_h = \sum_i W_{ih}.X_i - \theta_h \tag{6}$$

$$H_h = f(net_h) = \frac{1}{1 + e^{-net_k}} \tag{7}$$

Equation (7) is the computation of the neural network between the input layer (X) and the hidden layer (H) where each input node is represented as Xi and the node in the hidden layer is represented as $h \epsilon H$. A sigmoid function $f(net_h)$ is used to transfer data for each node in the hidden layer. Equation (8) shows the computation of the weighted sum (net_j), for each node in each output layer. Theta (θ_j) is the bias for each node, and W_{hj} is the weight for each node from the hidden layer to the output layer.

$$net_j = \sum_i W_{hj}.H_h - \theta_j \tag{8}$$

$$Y_j = f(net_h) = \frac{1}{1 + e^{-net_j}} \tag{9}$$

The output value (Y_j) between the hidden layer (H) and the output layer (Y) is calculated using equation (9). The sigmoid function $f(net_h)$ is used to transfer the data from output node Y_j. This baseline model is a binary classification with two outputs: 1 if the system predicts its user rating is the same as the classifiers rating, or 0 for the opposite rating. After the baseline classifier calculates all the possible user ratings from its input vector, the system uses an ensemble model with a vote (Bayes optimal classifier) to determine the user rating as the system output value.

IV. Experiments

In this section, our research experiments are explained and illustrated. For the deep learning model, the system uses 70 percent of the data for training and 30 percent for testing. In the experiments, three datasets from YELP, Amazon, and Trip-advisor are used to test the system. For the deep learning hyper-parameter setting, we use a grid search to find the best combination. We use five hidden layers. Every hidden layer node is 50, 40, 30, 20, and 10. We also use a dropout rate of 0.4. The activation function is ReLu. Regularization L2 also implemented in the system. The regularization has a learning rate of 0.0001 and a weight decay of 0.000001.

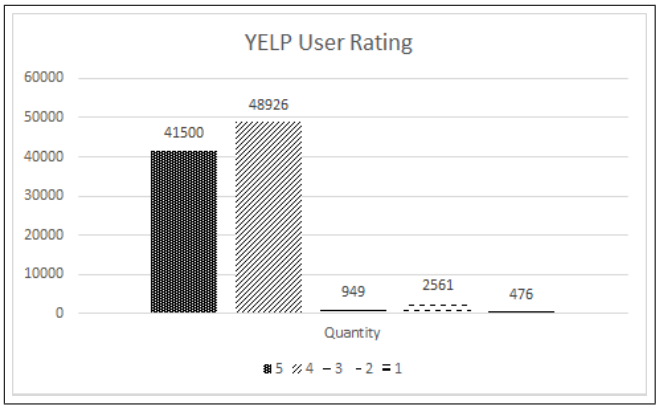

Fig. 6. The Imbalance YELP's User Rating Dataset

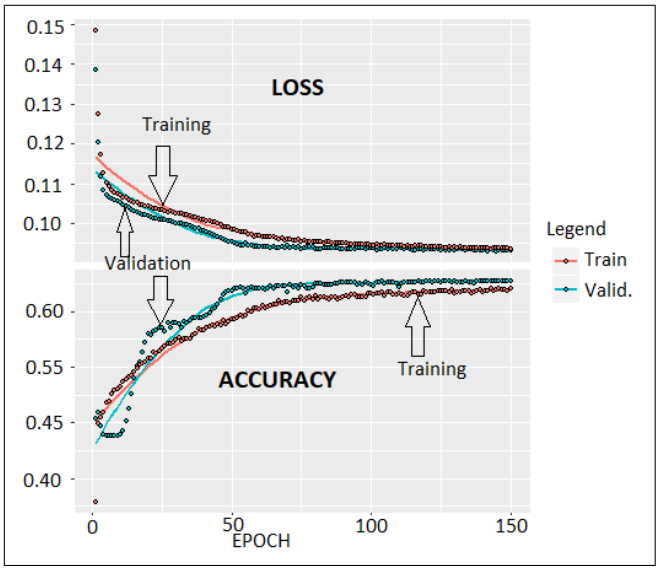

Fig. 7. The Training Results on YELP Dataset

TABLE I
COMPARISON OF MSE LOSS AND ACCURACY TRAINING MODELS ON YELP DATASET

Methods	MSE Loss	Accuracy
LibSVM	0.1489	0.6277
MLP	0.0924	0.6232
DBN	0.0937	0.6236
IDBN	**0.0918**	**0.6339**

A. Experiments of YELP Dataset

To simplify our dataset, we use the YELP datasets restaurant category. The dataset consists of 94,412 user comments giving restaurant ratings. Figure 6 shows the distribution of YELPs user rating dataset.

Figure 6 shows that the user rating is skewed to good and very good (rating 4 and rating 5). Rating 5 has 41,500 votes, 4 has 48,926 votes, while 3 has a mere 949 votes. The bad and very bad ratings have few votes. Rating 2 has 2,561 votes and 1 just 476 votes.

Figure 7 shows the system training model. The training model converges after 48 iterations, with its best MSE loss value of 0.093. The systems best accuracy value is 0.6255. In the testing phase, we test our system with an accuracy metric. The systems best MSE loss metric is 0.0918 and the best accuracy is 0.6339.

We also compare IDBN model with other baseline methods such as LibSVM, Multi-Layer Perceptron (MLP), and traditional DBN. Table I shows comparison results between the methods. Table I shows that our proposed method outperforms other baseline methods. The testing accuracy of the model and that of DBN is similar by MSE loss. Our MSE loss has the best training value with 0.0918, but this value is better than DBNs MSE loss value by only 1%. Our proposed method has a better accuracy value than the DBN. Our proposed method also outperforms the other two baseline methods with a small margin. Because the training phase uses a random value, this model likely fits only the good and very good ratings. This is because of the proportion of bad and very bad ratings is much smaller than that of the good and very good ratings.

TABLE II
COMPARISON BETWEEN DBN AND IDBN IN GOOD AND BAD TESTING DATA ON YELP DATASET

Methods	Rating Testing	Accuracy
DBN	Only Good and Very Good	0.6282
IDBN	Only Good and Very Good	0.6309
DBN	Only Bad and Very Bad	0.6236
IDBN	Only Bad and Very Bad	0.7290

Table II shows the accuracy results for the testing data for good and bad rating products. For the DBN, because of the imbalanced dataset it looks good when we use all the data because of the proportion of good and very good ratings is higher than the proportion of bad and very bad ratings. When we test with the bad ratings only, the accuracy is significantly lower because the DBN model does not fit this training model. Our system has greater sustainability for this kind of challenge, because of the deep learning ensemble model is trained to be fitted with imbalance data. The accuracy differences between DBN and our proposed model exceeds 10%, a significant improvement in predicting bad ratings from an imbalanced dataset.

B. Experiments of Amazon Dataset

The Amazon Dataset consists of 55,637 user comments for products in the camera category from 7,673 users. Figure 8 shows that the user comment distribution in the Amazon dataset is more balanced than that of the YELP dataset. Most users give very good ratings (rating 5, 29,493 votes). Rating 3, neutral has 14,099 votes, while Good (rating 4) has 3,561 votes. Rating bad (rating 2) has 5,191 votes and very bad (rating 1) has 3,293 votes.

Figure 9 shows the system training results. The model converges after 25 iterations. Its best MSE Loss value is 0.0736 and its best accuracy value is 0.7455. In the training phase, the deep learning model does not have the over-fitting problem as long as the validation loss value is lower than the training loss value. The value loss of the training model is always higher than the validation model because we apply the regularization L2 to the parameter setting. The L2 applies a loss penalty and ensures that the loss value is higher than the validation loss value.

Table III shows the comparison results between methods on Amazon dataset. Table III shows that the DBN has the best MSE Loss for training and the highest accuracy for the training dataset. This result is the opposite of our result for the YELP dataset. Although DBN has the best

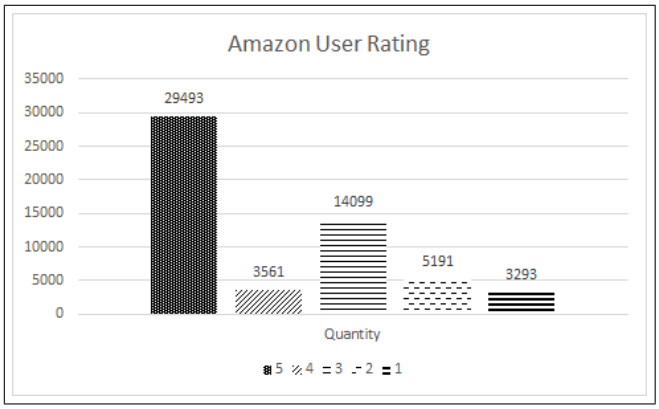

Fig. 8. The Imbalance Amazon's User Rating Dataset

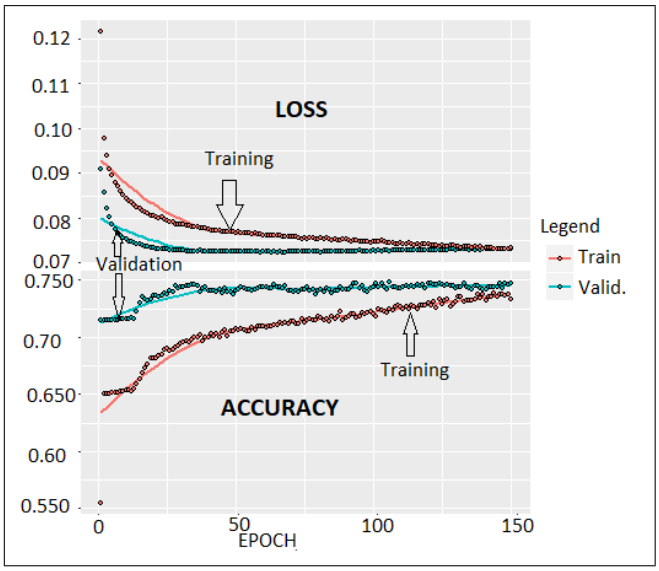

Fig. 9. Training Results on Amazon Dataset

TABLE III
COMPARISON OF MSE LOSS AND ACCURACY TRAINING MODELS ON AMAZON DATASET

Methods	MSE Loss	Accuracy
LibSVM	0.1774	0.5563
MLP	0.1208	0.5383
DBN	**0.0710**	**0.7585**
IDBN	0.0736	0.7455

MSE Loss and Accuracy value, IDBN is nearly as good as DBN. The differences do not exceed 1% in MSE loss and accuracy value, which is not significant. IDBN outperforms the LibSVM and MLP methods in MSE Loss and accuracy values. The LibSVMs MSE Loss value is 0.1774, and its best accuracy value is 0.5563. MLPs MSE Loss value is 0.1208, which is better than the LibSVM Loss value. MLPs accuracy value is 0.5383, lower than the LibSVM accuracy value. DBN exhibits the best MSE loss value, 0.0710, better than IDBN. Its MSE Loss value of 0.0736, and DBN accuracy value of 0.07585 are better than IDBN accuracy value of 0.7455.

In Table IV, we compare our method with the traditional DBN in predicting user ratings in the good and very good rating and bad and very bad rating. Table III shows that DBN obtains better results in the training phase but when we test the accuracy for the good and bad rating separately,

TABLE IV
COMPARISON BETWEEN DBN AND IDBN IN GOOD AND BAD TESTING DATA ON AMAZON DATASET

Methods	Rating Testing	Accuracy
DBN	Only Good and Very Good	0.9596
IDBN	Only Good and Very Good	**0.9985**
DBN	Only Bad and Very Bad	0.4471
IDBN	Only Bad and Very Bad	**0.6653**

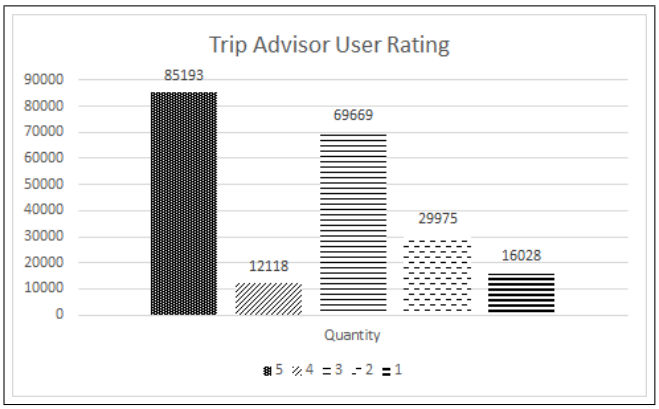

Fig. 10. Imbalance Trip-Advisor's User Rating Dataset

the model returns lower accuracy results, especially when the testing data is for the bad rating. It makes sense because DBN data is trained to fit good rating data better. When the testing data is bad rating data, the model has lower accuracy. Our proposed method returns an accuracy value of 0.6653 for bad rating data, and 20% higher accuracy than DBN. Those values show our model sustainability is better than that of the DBN model.

C. Experiments of Trip-Advisor Dataset

Trip-advisor Dataset consists of 212,983 user comments for hotel businesses around the world from 12,773 users. Figure 10 shows that the user comment distributions in the Trip-advisor dataset is more balanced than that of the YELP dataset, shown in Figure 6. The Trip-advisor dataset has a distribution of user ratings similar to that of the Amazon Dataset. From Figure 10, we find that Rating 5, very good, has 85,193 votes, rating 3, the neutral impression, has 69,669 votes, and the good rating (rating 4) has 12,118 votes. The rating bad (rating 2) has 29,975 votes and very bad (rating 1) has 16,028 votes.

Figure 11 shows the system training results for the Trip-advisor dataset. The model converges after 75 iterations. Its best MSE Loss value is 0.2052 and its best accuracy value is 0.4910. In the training phase, the deep learning model does not have an over-fitting problem as long as the validation loss value is lower than the training loss value. The value loss of the training model is always higher than the validation model. This is possible because we apply the regularization L2 in the parameter setting. L2 applies a loss penalty and ensures that the loss value will be higher than the validation loss value.

Table V shows results for different methods using the Trip-advisor dataset. Table V shows that DBN has the best MSE Loss for training and the highest accuracy for the training dataset, the opposite of our results with the Yelp dataset. Though DBN has the best MSE Loss value and

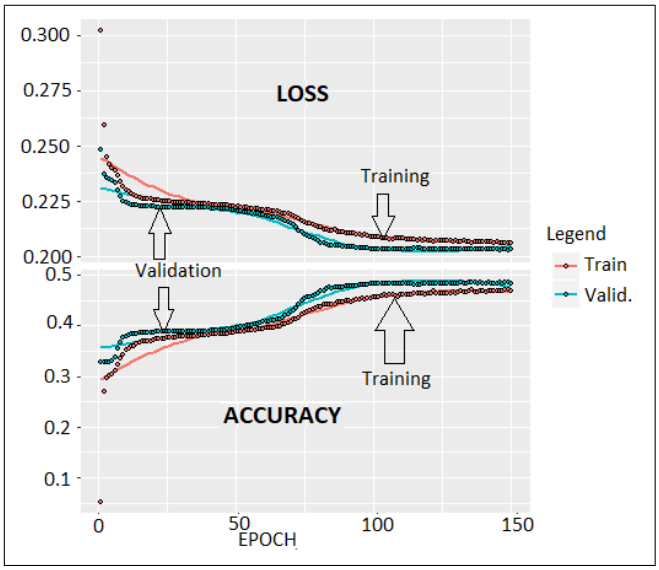

Fig. 11. Training Results on Trip-advisor Dataset

TABLE V
MSE LOSS AND ACCURACY COMPARISON METHODS ON TRIP-ADVISOR DATASET

Methods	MSE Loss	Accuracy
LibSVM	0.1998	**0.5002**
MLP	0.1230	0.4838
DBN	**0.1216**	0.4923
IDBN	0.2052	0.4910

TABLE VI
COMPARISON BETWEEN DBN AND IDBN IN GOOD AND BAD TESTING DATA ON TRIP-ADVISOR DATASET

Methods	Rating Testing	Accuracy
DBN	Only Good and Very Good	**0.4757**
IDBN	Only Good and Very Good	0.4235
DBN	Only Bad and Very Bad	0.3429
IDBN	Only Bad and Very Bad	**0.4695**

LibSVM has the best accuracy value, our proposed methods are nearly as good as the DBN results. The difference in MSE loss and the accuracy value is less than 1%. Our proposed method outperforms MLP methods in MSE Loss and accuracy values. LibSVMs MSE Loss value is 0.1998 and its best accuracy value is 0.5002. MLPs MSE Loss value is 0.1230, better than LibSVMs Loss value. MLPs accuracy value is 0.4838, lower than LibSVMs value. DBN has the best MSE loss value at 0.1216, which is better than our proposed methods MSE Loss value of 0.2052. DBN has an accuracy value of, 0.4923 better than our proposed methods value of 0.4910.

In Table VI, we compared our proposed method to traditional DBN for predicting the user ratings good, very good, bad, and very bad. Table V shows that DBN has better results in the training phase, but when we test the accuracy in for the good and bad rating separately, the model returns lower accuracy results, especially when testing bad rating data. The DBN data is trained to better fit good rating data. Thus, when testing bad rating data, the model has lower accuracy. Our proposed method returns an accuracy value of 0.4695 for bad rating testing data and has an accuracy value of 10% higher than DBN for bad rating data. Our model's sustainability is better than that of the DBN model.

D. Discussion

In Neural network, the training process is like a black box. We could not fully understand the process inside. But we know for sure if the single model is trained for many classes, nodes which are inside the model will tolerate each other to be fitted with all the classes.

In imbalance dataset, nodes which are inside the model will be fitted more into the majority class. Table II shows this case in YELP dataset, Table IV in Amazon dataset and Table VI in Trip Advisor dataset. We can find in Table II, the accuracy of the majority class "Good" and "Very Good" rating for the traditional DBN which gets almost the same accuracy with our proposed method. However, when we test with "Bad" and "Very Bad" rating which are minority class, the traditional DBN suffer from getting high accuracy. In contrast, our proposed method does not suffer from this problem as we train our model fit with the minority class.

Table IV and Table VI also show similar results with Table II. We compare with others dataset to test our model. We implemented the ensemble model in this case to let our model to learning specific class. Ensemble model could let the model train independently and separate the train process into majority and minority class.

V. CONCLUSION

We propose a deep learning system incorporating ensemble methods to predict user ratings from user comments in product reviews. From the experiments, our method outperforms other baseline methods. When we use imbalanced datasets, our proposed methods have greater sustainability in accuracy metrics than traditional baseline deep learning methods. In the present work, we consider only Bayes optimal classifiers as our votes for determining user ratings.

In the future, other parameters such as social influences and emotional factors that affect user ratings and user comments can be taken into consideration to enhance the accuracy of the system. We also aim to improve the system by using a data sampling method to reproduce the data in order to get more balance class and test the novelty with cost-sensitive learning algorithm.

ACKNOWLEDGMENT

This paper is supported by the Ministry of Science and Technology, Taiwan, with project number: MOST-106-2221-E-324-025 and MOST-106-2218-E-324

REFERENCES

[1] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[3] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, pp. 504–507, 2006.

[4] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006. [Online]. Available: http://dx.doi.org/10.1162/neco.2006.18.7.1527

[5] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *CoRR*, vol. abs/1207.0580, 2012. [Online]. Available: http://arxiv.org/abs/1207.0580

[6] C. L. P. Chen and Z. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 1, pp. 10–24, 2018.

[7] S. Maldonado and J. Lpez, "Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for svm classification," *Applied Soft Computing*, vol. 67, pp. 94 – 105, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1568494618301108

[8] L. Yu, R. Zhou, L. Tang, and R. Chen, "A dbn-based resampling svm ensemble learning paradigm for credit classification with imbalanced data," *Applied Soft Computing*, vol. 69, pp. 192 – 202, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1568494618302400

[9] A. Nayal, H. Jomaa, and M. Awad, "Kerminsvm for imbalanced datasets with a case study on arabic comics classification," *Engineering Applications of Artificial Intelligence*, vol. 59, pp. 159 – 169, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0952197617300027

[10] L. Bao, C. Juan, J. Li, and Y. Zhang, "Boosted near-miss under-sampling on svm ensembles for concept detection in large-scale imbalanced datasets," *Neurocomputing*, vol. 172, pp. 198 – 206, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231215006098

[11] M. R. Silvestre and L. L. Ling, "Pruning methods to mlp neural networks considering proportional apparent error rate for classification problems with unbalanced data," *Measurement*, vol. 56, pp. 88 – 94, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0263224114002796

[12] Z.-Q. Zhao, "A novel modular neural network for imbalanced classification problems," *Pattern Recognition Letters*, vol. 30, no. 9, pp. 783 – 788, 2009, advanced Intelligent Computing Theory and Methodology. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865508001979

[13] O. Araque, I. Corcuera-Platas, J. F. Snchez-Rada, and C. A. Iglesias, "Enhancing deep learning sentiment analysis with ensemble techniques in social applications," *Expert Syst. Appl.*, vol. 77, no. C, pp. 236–246, 2017. [Online]. Available: https://doi.org/10.1016/j.eswa.2017.02.002

[14] R. Xia and C. Zong, "A pos-based ensemble model for crossdomain sentiment classification," in *in Proceedings of the 5th International Joint Conference on Natural Language Processing*, 2010.

[15] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Editorial: Special issue on learning from imbalanced data sets," *SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 1–6, 2004. [Online]. Available: http://doi.acm.org/10.1145/1007730.1007733

[16] C. Elkan, "The foundations of cost-sensitive learning," in *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, ser. IJCAI'01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 973–978. [Online]. Available: http://dl.acm.org/citation.cfm?id=1642194.1642224

[17] Q. Zou, S. Xie, Z. Lin, M. Wu, and Y. Ju, "Finding the best classification threshold in imbalanced classification," *Big Data Research*, vol. 5, pp. 2 – 8, 2016, big data analytics and applications. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2214579615000611

[18] T. Voigt, R. Fried, M. Backes, and W. Rhode, "Threshold optimization for classification in imbalanced data in a problem of gamma-ray astronomy," *Advances in Data Analysis and Classification*, vol. 8, no. 2, pp. 195–216, 2014. [Online]. Available: https://doi.org/10.1007/s11634-014-0167-5

[19] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463–484, 2012.

[20] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 20–29, 2004. [Online]. Available: http://doi.acm.org/10.1145/1007730.1007735

[21] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *J. Artif. Int. Res.*, vol. 16, no. 1, pp. 321–357, 2002. [Online]. Available: http://dl.acm.org/citation.cfm?id=1622407.1622416

[22] A. Uyar, A. Bener, H. N. Ciray, and M. Bahceci, "Handling the imbalance problem of ivf implantation prediction," *IAENG International Journal of Computer Science*, vol. 37, no. 2, pp. 164–170, 2006.

[23] K. Savetratanakaree, K. Sookhanaphibarn, S. Intakosum, and R. Thawonmas, "Article: Borderline over-sampling in feature space for learning algorithms in imbalanced data environments," *IAENG International Journal of Computer Science*, vol. 43, no. 3, pp. 363–373, 2016.

[24] B. Zadrozny and C. Elkan, "Learning and making decisions when costs and probabilities are both unknown," in *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '01. New York, NY, USA: ACM, 2001, pp. 204–213. [Online]. Available: http://doi.acm.org/10.1145/502512.502540

[25] P. Domingos, "Metacost: A general method for making classifiers cost-sensitive," in *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '99. New York, NY, USA: ACM, 1999, pp. 155–164. [Online]. Available: http://doi.acm.org/10.1145/312129.312220

[26] B. Zadrozny, J. Langford, and N. Abe, "Cost-sensitive learning by cost-proportionate example weighting," in *Proceedings of the Third IEEE International Conference on Data Mining*, ser. ICDM '03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 435–. [Online]. Available: http://dl.acm.org/citation.cfm?id=951949.952181

[27] B. Liu, *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012.

[28] L.-C. Yu, J.-L. Wu, P.-C. Chang, and H.-S. Chu, "Using a contextual entropy model to expand emotion words and their intensity for the sentiment classification of stock market news," *Know.-Based Syst.*, vol. 41, pp. 89–97, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.knosys.2013.01.001

[29] M. Hagenau, M. Liebmann, and D. Neumann, "Automated news reading: Stock price prediction based on financial news using context-capturing features," *Decision Support Systems*, vol. 55, no. 3, pp. 685 – 697, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167923613000651

[30] T. Xu, Q. Peng, and Y. Cheng, "Identifying the semantic orientation of terms using s-hal for sentiment analysis," *Knowledge-Based Systems*, vol. 35, pp. 279 – 289, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705112001074

[31] A. D'Andrea, F. Ferri, P. Grifoni, and T. Guzzo, "Article: Approaches, tools and applications for sentiment analysis implementation," *International Journal of Computer Applications*, vol. 125, no. 3, pp. 26–33, 2015, published by Foundation of Computer Science (FCS), NY, USA.

[32] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093 – 1113, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2090447914000550

[33] A. Bruun and S. Ahm, "Mind the gap! comparing retrospective and concurrent ratings of emotion in user experience evaluation," in *Human-Computer Interaction – INTERACT 2015*, J. Abascal, S. Barbosa, M. Fetter, T. Gross, P. Palanque, and M. Winckler, Eds. Cham: Springer International Publishing, 2015, pp. 237–254.

[34] P. J. Lang, "Behavioral treatment and bio-behavioral assessment: Computer applications," in *Technology in mental health care delivery systems*, J. B. Sidowski, J. H. Johnson, and T. A. Williams, Eds. Norwood, NJ: Ablex, 1980, pp. 119 – 137.

[35] D. Kollias, G. Marandianos, A. Raouzaiou, and A.-G. Stafylopatis, "Interweaving deep learning and semantic techniques for emotion analysis in human-machine interaction," in *Proceedings of the 2015 10th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP)*, ser. SMAP '15. Washington, DC, USA: IEEE Computer Society, 2015, pp. 1–6. [Online]. Available: http://dx.doi.org/10.1109/SMAP.2015.7370086

[36] G. E. Hinton, "Deep belief networks," *Scholarpedia*, vol. 4, p. 5947, 2009.

Hendry was born in Negara, Bali, Indonesia in 1983. He received the B.S. in Information Technology from Technology School of Surabaya in 2005 and M.S degree in Information Technology from 11 November Institute of Technology, Surabaya in 2009, and Ph.D in Department of Information Management Chaoyang University of Technology, Taichung, Taiwan. From 2012-2014 he was Director in Business and Technology Incubator Satya Wacana Christian University. His research interest include domain ontology, recommendation system, knowledge engineering, the applications of Artificial Intelligent

Rung-Ching Chen Rung-Ching Chen received the B.S. degree from department of electrical engineering in 1987, and the M. S. degree from the institute of computer engineering in 1990, both from National Taiwan University of Science and Technology, Taipei, Taiwan. In 1998, he received the Ph.D. degree from the department of applied mathematics in computer science sessions, National Chung Tsing University. He is now a distinguished professor at the Department of Information Management, Taichung, Taiwan. He was a Department Chair of Information Management in CYUT, Chaoyang University of Technology, between 2005 and 2007. He has been a Dean of Informatics College of CYUT from 2007 to 2015. He was the Execution Director of Institution Research Office during Dec. 2015 and Jan. 2017 in CYUT. He also the president of Taiwanese Association for Consumer Electronics during December 2012 and December 2014. He has been a Fellow of IET since July 2011. He was a visiting Professor at University of Central Florida, U.S.A. in 2012 and a visiting researcher at the University of Aizu, Japan, in 2017. He got a PMP(Project Management Professional) certification in 2013. His research interests include networks technology, domain ontology, pattern recognition and knowledge engineering, IoT and data analysis, the applications of Artificial Intelligent.