International Journal of Innovative Computing, Information and Control
Volume **14**, Number **3**, June **2018**

ICIC International ©2018 ISSN 1349-4198
pp. 1141–1149

USING DEEP LEARNING TO LEARN USER RATING FROM USER COMMENTS

Hendry[1,2], Rung-Ching Chen[1,*], Li-Hua Li[1] and Qiangfu Zhao[3]

[1]Department of Information Management
Chaoyang University of Technology
168, Jifeng E. Rd., Wufeng District, Taichung 41349, Taiwan
{ s10314905; lhli }@cyut.edu.tw; *Corresponding author: crching@cyut.edu.tw

[2]Faculty of Information Technology
Satya Wacana Christian University
Jl. Diponegoro 52-60, Salatiga 50711, Indonesia
hendry@staff.uksw.edu

[3]Department of Computer Science
The University of Aizu
Tsuruga, Ikki-machi, Aizu-Wakamatsu City, Fukushima 965-8580, Japan

Received August 2017; revised December 2017

Abstract. *Current recommender systems usually take scores or ratings for data analysis. Although user rating for a certain item (product) is easier to obtain, rating alone cannot tell us what the users are thinking about. User comments are essential for recommender systems because they include various types of emotional information that may influence the correctness or precision of the recommendation. To obtain feasible recommendations, it is important to improve the accuracy of user rating. The goal of this research is to propose a deep learning model to process the emotion terms of user comments and to generate feasible user rating for the recommendation. Our deep learning method utilizes the GloVe algorithm to create word vector which will be used as the input for the training model. Our proposed method has three major steps. The first step is to transform user comments into word feature vectors; the second step is to learn user emotion from user comments based on the Restricted Boltzmann Machine (RBM); the third step is to learn the user rating by using a back-propagation neural network based on the output of the RBM. The results of the proposed deep learning model can be applied for recommendation in social network environment.*
Keywords: Deep learning, Recommendation system, Emotion classification

1. **Introduction.** In recent years, recommender systems are gradually escaping from traditional single recommendation methodology. Many studies have shown that recommender systems can be combined with semantic web reasoning [1]. Related technologies such as ontologies are also incorporated to generate intelligent recommendations [2]. In combination with mobile devices, digital contents can be transmitted to users easier than before. Recent studies also have shown that the establishment of personal ontology can help the reasoning or inference for ontology development. By analyzing the user's characteristics and the usage records, a user's ontology can be constructed. This kind of ontology can be adjusted at any time by using the historical records to provide better services for users [1-3]. It is, therefore, the major research topics of combining ontology, big data and recommender system in the field of information social sciences.

X. Qian et al. [4] proposed a method to calculate a recommender system based on user's interests from social circles. They used the inferred trust from social circles to recommend items. D. Ahn et al. [5] proposed personalized recommender systems based on the familiarity of the same topics among users in a social network. This approach considers how the user likes and comments on Facebook posts. J. Wu et al. [6] proposed prediction of user ratings from texts incorporating semantic graph. S. Liang et al. [7] proposed a singularity-based method for measuring user similarity. They only use user rating to calculate the similarity between users.

Y. Zheng et al. [8] used emotional context in recommender systems. They pointed out "emotions are crucial for user's decision making in the recommendation process". User rating could be reconsidered as user emotion. With the current knowledge, there are rare of researches regarding user's emotion for new recommender system. In another side, automatic recognition of text emotion is getting more and more attention currently [9]. For example, a sentiment system can read the news and tell the news is negative, positive or neutral [1]. S. M. A. Masum et al. [11] recently have developed "SenseNet" that is a linguistic tool to discover polarity values of a word. In their approach, with the help of WordNet, their method can assign a numerical value for some verbs, adjectives, and adverbs. In other words, they detect their polarity values using ConceptNet.

Deep learning is one of the methodologies that can be used to calculate the emotional factors. In [12], they used deep learning to recognize emotion based on the recognition of speech emotion. In [13], deep learning is utilized to recognize emotion in multimodal, such as the combination of video and audio from users. In [14], deep learning is utilized to recognize emotion in images, vision, and videos to improve the emotion classification and cross-modal retrieval. K. Dimitris et al. [13] utilized deep learning to analyze user's emotion based on human-machine interaction. The aims of previous researchers are to improve the classification results of emotion factors via vision, images, or audio.

Many papers proposed recommender system based on user interest from social circles but user emotion is not considered [4-7]. Some papers proposed recommender system based on emotion from images, vision and videos of users [8,9,12-14]. The drawback is hard to collect user emotions through user's image, vision and video for each user.

In this paper, we propose an alternative approach which utilizes deep learning method by analyzing users' comments. We utilize the GloVe algorithm to generate word vector to learn user emotions, which is more straightforward to compare it with previous methods [8,9,12-14]. We assume that emotion is one of the factors which has a significant impact on recommender systems and can make the recommendation more accurate. Users' comments are processed from texts into words and will be represented concerning vectors. These vectors will be used as input to deep learning model and the classification of user's emotion will be learned. Based on these emotion factors, the proposed system will predict the user rating for the recommendation. Our proposed model can be used to predict the recommendation items through personalized recommender system.

The remainders of the paper are organized as follows. Section 2 gives problem definition and preliminary preparation. Section 3 proposes a new method for predicting user rating based on deep learning and the GloVe algorithm. Section 4 discusses the experimental results to verify the proposed method. Finally, in Section 5, we give conclusions and some topics for future work.

2. **Problem Definition and Preliminary.** As stated in [8], emotion could affect user's decision on giving a rating for items. The dataset consists of a set of users, $U = \{u_1, u_2, \ldots, u_m\}$ and a set of businesses, $Bu = \{bu_1, bu_2, \ldots, bu_n\}$. Rating for each business bu with user u is expressed by ratings $R = R[u, bu]_{m\times n}$. This matrix $R_{u,bu}$ denotes the rating of user u on business bu. $R_{u,bu}$ could be any number in the range of 1 to 5,

as we collected from the database. In this paper, we will represent rating value by using binary value as output in our deep learning model.

This research proposes an alternative approach to calculate user rating based on user comments. For every comment from the business review, we will obtain a set of comments C, where $C = \{c_{1,1}, c_{1,2}, \ldots, c_{m,n}\}$, c represents business bu which is commented by user u. GloVe [15] has proposed an algorithm to represent word into word vector. GloVe's work was based on semantically relations between words, and these word vectors were calculated in terms of the matrix. In [15], it already demonstrated that GloVe algorithm outperforms the algorithm of skip-gram model, Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA).

GloVe algorithm is to generate word vectors for all user comments which are represented as $Ve = [Ve_{u,bu}]_{k,l}$, where k is the dimension of vector and l is the total number of user comments u for business bu. Matrix $Ve_{u,bu}$ is the word vector representing the comment of user u to business bu. We use these word vectors, v, as input for our deep learning model. Our deep learning model is based on Restricted Boltzmann Machine (RBM). RBM is a deep learning structure which consists of the visible layer and hidden layers with fully connected weights between layers (see Figure 1). RBM uses binary value for each node for both the visible and hidden layers. To transfer real value data into RBM learning model we must transform real value data using real-valued Gaussian function into binary value [16].

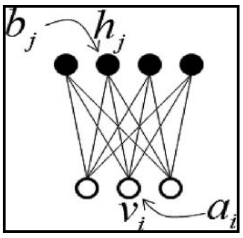

Figure 1. RBM model

In Figure 1, standard RBM units consist of visible layer v_i, with hidden layer h_j and matrix of weights, $W = [w_{i,j}]_{m\times n}$, where W connects v_i with h_j, respectively. RBM bias is represented by a_i for the visible layer and b_j for hidden layer. The energy function for a configuration (v, h) is defined as Equation (1).

$$E_{(v,h)} = -\sum\nolimits_{ij} v_i W_{ij} h_j - \sum\nolimits_i a_i v_i - \sum\nolimits_j b_j h_j \tag{1}$$

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

$$P_{(v)} = \frac{1}{Z}\sum\nolimits_h e^{-E_{(v,h)}} \tag{2}$$

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.

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

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

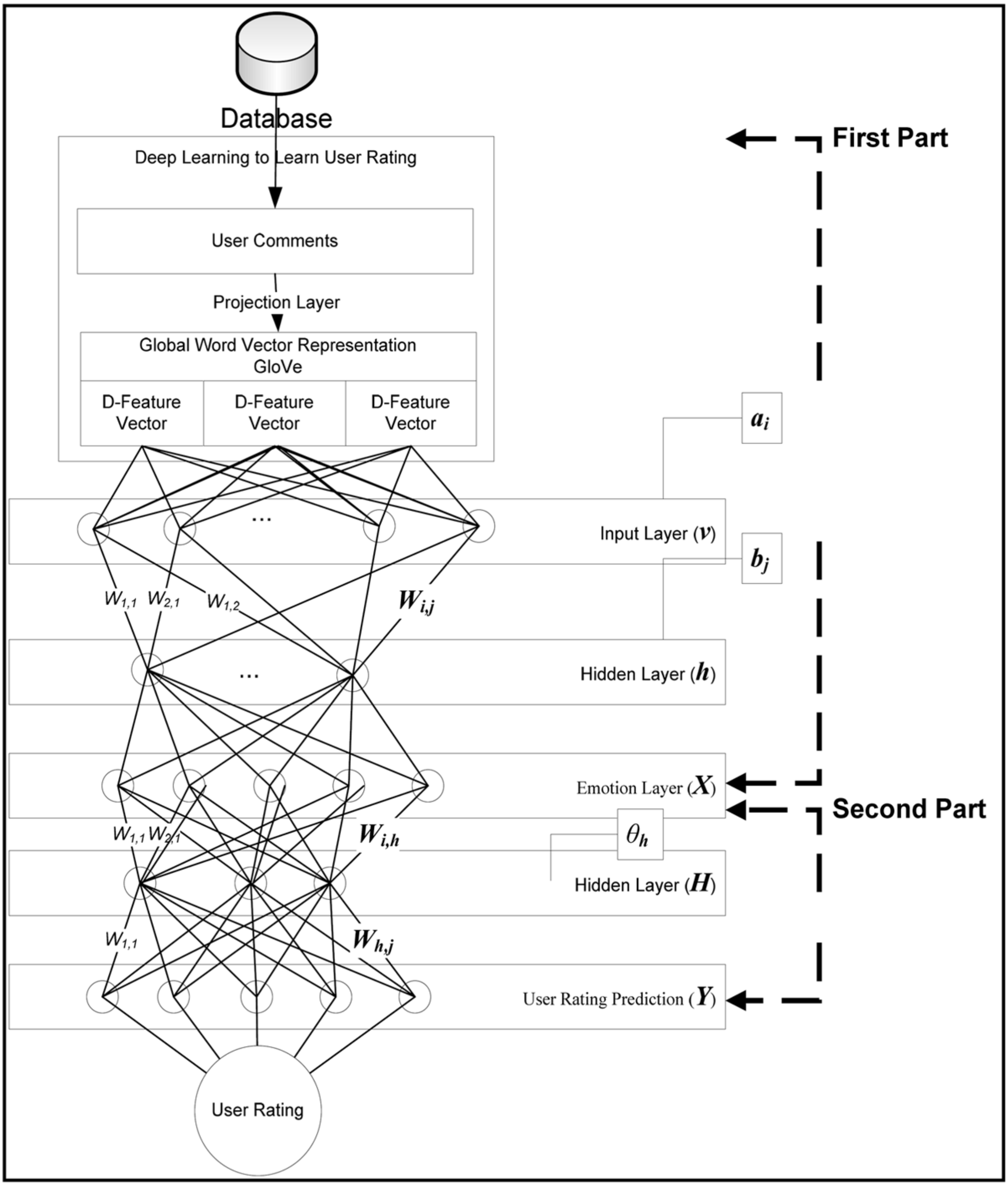

Figure 2. The proposed system architecture

3. **The Proposed Methodology.** This section presents our approach to learn user rating by incorporating the vector representation of the word, deep learning and tradeoff of emotions in users' comments. Figure 2 shows the proposed system architecture. The first part is to represent user comments by using word-vector. The GloVe algorithm is utilized to do the word-vector representation. After the word-vector representation is generated, our system will execute the learning process to perform the emotion classification from user comments. The main steps for this part are as follows:

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

$$p(h_j = 1|v) = \sigma\left(\sum\nolimits_i W_{ij}v_i + b_j\right) \tag{4}$$

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

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

$$p(v_i'|h) = \sigma\left(\sum\nolimits_j W_{ij}h_j + a_i\right) \tag{5}$$

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 input layer is reconstructed into new input vector layer from RBM learning.

The second part of our proposed deep learning model uses emotion classification as inputs to learn the user rating. Equation (6) shows the computation of weighted sum net_h, for each node in the hidden layer. Theta (θ_h) is the bias for each hidden node and W_{ih} represents the weight of the connection between input node X_i and hidden node h.

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

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

Equation (7) is the computation of neural network between input layer (X) and hidden layer (H) where each input node is represented as X_i and the node in the hidden layer is represented as h, $h \in H$. Sigmoid function $f(net_h)$ is used to transfer data to each node in the hidden layer.

Equation (8) shows the computation of weighted sum net_j, for each node in each output layer. Theta (θ_j) is the bias for each node; W_{hj} is the weight for each node from hidden layer to output layer.

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

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

The output value (Y_j) between hidden layer (H) and the output layer (Y) is calculated using Equation (9), and sigmoid function $f(net_h)$ is used to transfer the data from output node Y_j.

4. **Experimental Results.** In this section, our research experiments are explained and illustrated. For our deep learning model, we use 70 percent data as for the training and use 30 percent data as the testing. We also use 50, 100, 150 and 200 data to compare with the training model. We use Yelp dataset academic challenge version fifth that consists of more than 1.5 million comments, 366 thousand users and 61 thousand of business. To limit our scope, we only take top 10 of business categories from the dataset for our experiments as shown in Table 1. Table 2 shows the summary of our dataset.

For the learning model, we set our vector dimension to 51, epoch = 9,000 − 15,000, hidden units = 20, and output units = 5. Figure 3 shows the result for our experiments using dataset 50. The best iterations model for 50 data at 8,997 iterations with train error = 0.03, and the best iteration model for 100 data at 8,964 iterations with train error = 0.034. We also conduct our experiments with 150 and 200 user comments and 200 user comments are shown in Figure 4. The best iteration model for 150 data at 10,121 iterations with train error = 0.191, and the best iteration model for 200 data at 13,509 iterations with train error = 0.147.

TABLE 1. Categories of business review from Yelp dataset

No	Categories	Number of Reviews
1	Restaurant	21,892
2	Shopping	8,919
3	Food	7,862
4	Beauty & Spas	4,738
5	Nightlife	4,340
6	Bars	3,628
7	Health & Medical	3,213
8	Automotive	2,965
9	Home Services	2,853
10	Fashion	2,566

TABLE 2. Statistics of dataset

Type	Attribute	Frequency
Users Data	User_id, name, friends, fans, votes, ...	366,715 items
Business Data	Business_id, name, city, categories, address, ...	61,184 items
Review Data	User_id, business_id, review_text, rating, ...	1,569,264 items

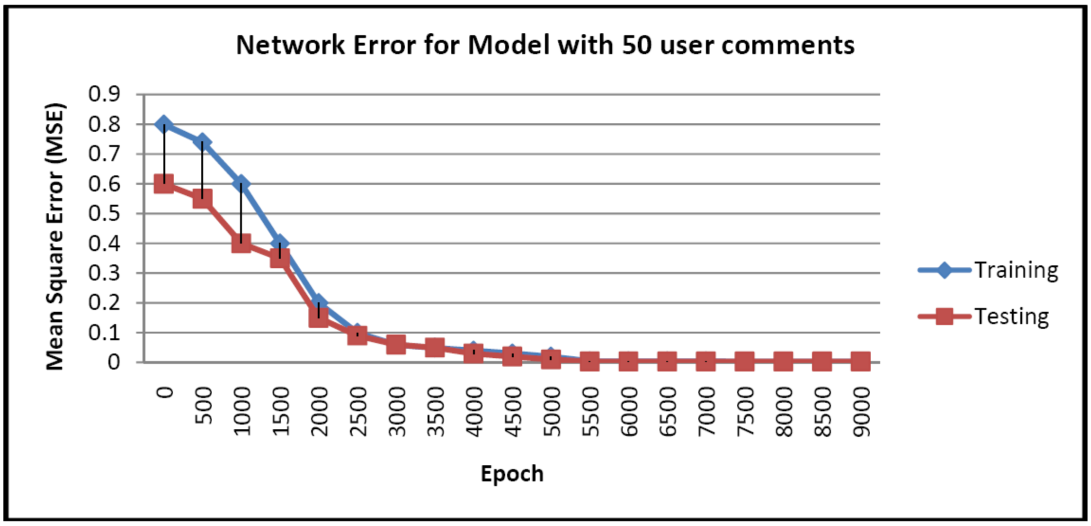

FIGURE 3. Result of training model for 50 user comments

Table 3 shows our system's accuracy for each data. We only show the best combination of deep learning parameters for each data. We also compare our model with another baseline model. Table 4 shows the comparison results of the Mean Squared Error (MSE) between our model and other models.

From Table 4 we could see that our proposed method outperforms the baseline methods. For smaller training sets our method is sigficantly better, and can fit better than other methods. For the training set with 200 user comments, the proposed method is still the best and MLP has the same training error.

5. **Conclusion and Future Work.** In this paper, we utilize deep learning to learn user rating through user comments. The method transforms user comments into word feature vector first. Next, Restricted Boltzmann Machine (RBM) is used to present user's emotion

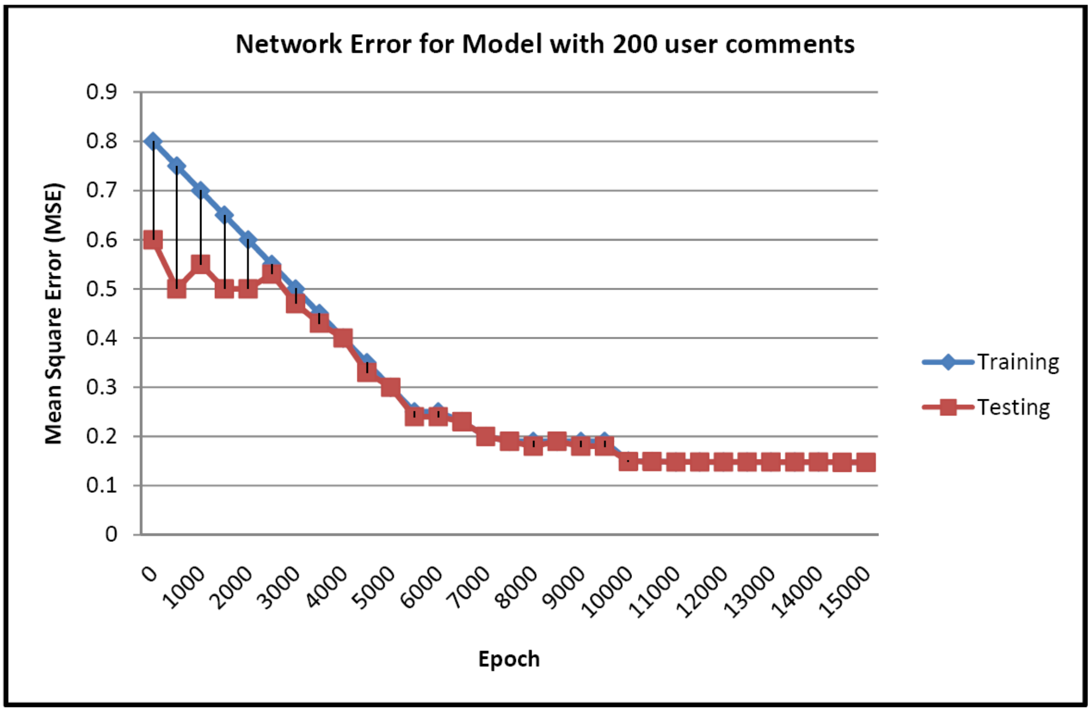

Figure 4. Result of training model for 200 user comments

Table 3. Results summary for each training data

Training Data	Deep Learning Model Parameter Settings	MSE	Accuracy
50 user comments	layers = $c(51, 20, 8, 5)$, bp.learnRate = .1, fineTuneFunction = "backpropagation", numEpochs = 9000, batchSize = 1, errorFunction = mseError, initialMomentum = 0.9	0.03	100%
100 user comments	layers = $c(51, 20, 8, 5)$, bp.learnRate = .1, fineTuneFunction = "backpropagation", numEpochs = 9000, batchSize = 1, errorFunction = mseError, initialMomentum = 0.9	0.034	96.96%
150 user comments	layers = $c(51, 20, 8, 5)$, bp.learnRate = .1, fineTuneFunction = "backpropagation", numEpochs = 15000, batchSize = 1, errorFunction = mseError, initialMomentum = 0.9	0.1907	85.23%
200 user comments	layers = $c(51, 20, 8, 5)$, bp.learnRate = .1, fineTuneFunction = "backpropagation", numEpochs = 15000, batchSize = 1, errorFunction = mseError, initialMomentum = 0.9	0.147	88.44%

from user comments. Finally, the system uses these data as the input to learn the user rating by utilizing a back-propagation neural network. From the experiments, our method outperforms other baseline methods. As for now we only use the local optimum method to determine the best training model. In the next step, we would like to incorporate ontology learning and big data analysis to process all our dataset (more than 1.5 million user comments), verify the proposed method, and propose methods for further improvement.

Table 4. MSE comparison for each method

Training Data	Methods	MSE of Testing Data
50 user comments	Decision Tree	2.15
	Multi-Layer Perceptron (MLP)	1.48
	Random Forest	0.29
	Propose Methods	**0.03**
100 user comments	Decision Tree	1.13
	Multi-Layer Perceptron (MLP)	0.14
	Random Forest	0.22
	Propose Methods	**0.034**
150 user comments	Decision Tree	1.33
	Multi-Layer Perceptron (MLP)	0.22
	Random Forest	0.25
	Propose Methods	**0.19**
200 user comments	Decision Tree	2.0049
	Multi-Layer Perceptron (MLP)	**0.14**
	Random Forest	0.27
	Propose Methods	**0.14**

Acknowledgment. This research is supported by Ministry of Science and Technology, Taiwan, with project number: MOST-106-2221-E-324-025 and MOST-106-2218-E-324-002.

REFERENCES

[1] Y. Sure, M. Erdmann, J. Angele, S. Staab, R. Studer and D. Wenke, OntoEdit: Collaborative ontology development for the semantic web, *Lecture Notes in Computer Science*, vol.2342, 2002.

[2] R.-C. Chen, Hendry and C. Y. Huang, A domain ontology in social networks for identifying user interest for personalized recommendations, *Journal of Universal Computer Science*, vol.22, no.3, pp.319-339, 2016.

[3] R. Mariam, Generating an excerpt of a service level agreement from a formal definition of non-functional aspects using OWL, *Journal of Universal Computer Science*, vol.20, no.3, pp.366-384, 2014.

[4] X. Qian, X. Liu, C. Zheng, Y. Du and X. Hou, Tagging photos using user's vocabularies, *Neurocomputing*, vol.111, pp.144-153, 2013.

[5] D. Ahn, T. Kim, S. J. Hyun and D. Lee, Inferring user interest using familiarity and topic similarity with social neighbors in Facebook, *Proc. of IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, Macau, China, pp.196-200, 2012.

[6] J. Wu, Z. Xuan and D. Pan, Enhancing text representation for classification tasks with semantic graph structures, *International Journal of Innovative Computing, Information and Control*, vol.7, no.5(B), pp.2689-2698, 2011.

[7] S. Liang, L. Ma and F. Yuan, A singularity-based user similarity measure for recommender systems, *International Journal of Innovative Computing, Information and Control*, vol.11, no.5, pp.1629-1638, 2015.

[8] Y. Zheng, R. D. Burke and B. Mobasher, The role of emotions in context-aware recommendation, *The 7th ACM Conference on Recommender Systems (RecSys 2013)*, Hong Kong, China, 2013.

[9] K. F. Yeung, Y. Yang and D. Ndzi, Context-aware news recommender in mobile hybrid P2P network, *The 2nd International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN)*, Liverpool, United Kingdom, pp.54-59, 2010.

[10] C. Strapparava and R. Mihalcea, Learning to identify emotions in text, *Proc. of the ACM Symposium on Applied Computing*, Fortaleza, Ceara, Brazil, pp.1556-1560, 2008.

[11] S. M. A. Masum, H. Prendinger and M. Ishizuka, Emotion sensitive news agent: An approach towards user centric emotion sensing from the news, *IEEE/WIC/ACM International Conference on Web Intelligence*, Fremont, CA, USA, pp.614-620, 2007.

[12] F. Wang, X. Ye, Z. Sun, Y. Huang, X. Zhang and S. Shang, Research on speech emotion recognition based on deep auto-encoder, *The 6th Annual IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems*, Chengdu, China, pp.308-312, 2016.

[13] K. Dimitris, M. George, R. Amaralys and S. Andreas-Georgios, Interweaving deep learning and semantic techniques for emotion analysis in human-machine interaction, *Semantic and Social Media Adaptation and Personalization*, Trento, Italy, pp.23-28, 2015.

[14] L. Pang, S. Zhu and C.-W. Ngo, Deep multimodal learning for affective analysis and retrieval, *IEEE Trans. Multimedia*, vol.17, no.11, pp.2008-2020, 2015.

[15] P. Jeffrey, S. Richard and D. M. Christopher, *GloVe: Global Vectors for Word Representation*, http://nlp.stanford.edu/pubs/glove.pdf, 2016.

[16] N. Phan, D. Dou, H. Wang, D. Kil and B. Piniewski, Ontology-based deep learning for human behavior prediction with explanations in health social networks, *Information Sciences*, vol.384, pp.298-313, 2016.