# Comment Evaluation by Combining Comment and Word Mutual Evaluation Method and LSTM Evaluation Method in Lecture Questionnaire

Nobuyuki Kobayashi*, Hiromitsu Shiina† and Takafumi Otani‡
* Faculty of Regional Management, Sanyo Gakuen University, Okayama, Japan, Email: koba_nob@sguc.ac.jp
†Faculty of Informatics, Okayama University of Science, Okayama, Japan Email: shiina@mis.ous.ac.jp
‡Graduate School of Informatics, Okayama University of Science, Okayama, Japan, Email: i18im02ot@ous.jp

***Abstract*—Many universities give free-description questionnaires to students to obtain feedback on faculty development (FD). When this is done, a proper analysis of the students' comments is necessary. The number of comments from the free description that can be acquired for the FD activities is often not very large. To evaluate a small amount of data with approximately 1500 comments here needs to be some improvement in the currently available evaluation methods. In this study, we propose a probability distribution for the evaluation. We also propose a method for mutually evaluating the words and the comments long with the LSTM evaluation method by using neural networks. However, these methods seem to have differences in accuracies between the estimated values of the closed tests and the unrated comments. Therefore, we apply two methods to the bootstrap method to estimate the unrated comments; we also propose a method to incorporate the comments into our solution.**

***Index Terms*—Comment evaluation, Free description analysis, Neural network, LSTM, Incorporating method**

## I. Introduction

Currently, many universities conduct faculty development (FD) activities to improve the faculty performance and research guidance capabilities; such developmental activities help to enrich the faculty and the graduate school education in Japanese universities. The FD activities use workshops, lectures, and symposiums to make classroom observations and lecture evaluations [1], [2]. According to the Ministry of Education survey, as of 2014, 752 national, public, and private universities, which amounts to 96% of all Japanese universities, have implemented student lecture evaluations [3]. In addition, the increase in class evaluations by students in each university showed that executing lecture questionnaires is an important FD activity. Typically, lecture questionnaires include multiple-choice items, that is, close-ended questions and free descriptions (i.e., open-ended comments). The quantitative evaluation of multiple-choice questions is relatively simple; however, the form of the questions is limited, and the answers are restricted to a predetermined set of answers. In contrast, open-ended questions that allow free-form answers are more flexible and may provide more personalized information. For universities aiming to improve the quality of education by implementing lecture questionnaires, honest student opinions relating to the faculty and the lectures are considered an important source of information. It is important to establish a method to automatically analyze the answers provided to open-ended questions. Some related methodologies have been widely researched. For example, natural language processing, which employs various techniques (such as morphological [4] and dependency analyses [5] to make human language understandable by the computer) has received considerable attention.

Other studies have investigated the classification of the evaluation text [6] and the sentiment analysis and have found that these techniques can be applied effectively for product reviews and freeresponse items in questionnaires [7], [8], [9]. Machine learning methods have achieved high document classification accuracy; other studies, such as Naive Bayes methods [10] and Support Vector Machines (SVM) [11], [12], have been used to classify the evaluation documents.

In this study, we focus on the free-response text (hereinafter, " comments ") in the lecture questionnaires. We employed polarity evaluation (based on positive and negative responses) to evaluate the comments and the specific words in the comments. Useful comments were extracted, and the faculty members and their lectures were evaluated based on these comments. We evaluated each comment based on the polarity values of the words in the comment and on the emotional terms dictionary created by Takamura et al. [13], [14].

However, this emotional terms dictionary was created based on texts, such as commentaries; therefore, discrepancies were evident among the target questionnaire words, the comment evaluations, and the polarity evaluations. When such differences are present, it is difficult to evaluate the comments and words using the emotional terms dictionary. For a free description of the lecture questionnaire, we propose two methods for mutually evaluating the words used for comments, we also use a method that involves the Long Short-Term Memory (LSTM)[15] of neural networks. In the comment evaluation of the mutual evaluation method, the first characteristic is to evaluate the rating estimates of the words and comments; these are repeated until the evaluation converges. The second characteristic is that the comments and words mutual evaluation method is used to calculate the ratings that are divided into stages by the probability distribution based on the mixed normal distribution.

However, a sentiment analysis that uses machine learning

for neural networks has been developed recently[16], [17]; this analysis uses LSTM. Therefore, in this research, we investigate the estimation of the ratings of lesson questionnaires by machine learning using the LSTM of neural networks. In addition, we propose two methods: the mutual evaluation method and the LSTM evaluation method. The LSTM evaluation method has better closed test accuracy. However, when estimating many unrated comments, the evaluation tends to be low. In the mutual evaluation method, the accuracy of the closed test is low, but the evaluation is not poor even when there are many unrated comments. As a result, LSTM has been given a low evaluation for comments having many words that were not included in the learning data. Therefore, we apply two methods to the bootstrap method to estimate the unrated comments and to propose a method to incorporate comments into the solution. We studied the difficulty ratings of English, Japanese, and Chinese with [18], [19] by using SVMs for the correlation of sentences and words. An SVM shows good accuracy for binary classification. However, the multiclass classification has room for improvement. Also, there is a tendency to weak data with high complexity, such as language information. The random forest [20] also thinks it has properties similar to SVM; therefore, we adopt a method that uses the probability distribution. Moreover, it is easier to synthesize by using the probability distribution. LSTM is in the process of development, and improving the LSTM composition makes it possible to evaluate the comment ratings and the word evaluations simultaneously. For this reason, we decided to use LSTM in this research.

## II. Questionnaire Data

This study used a single open question item from a lecture questionnaire administered at Okayama University of Science. The questionnaire was administered mid-course in the eighth term, which was the spring term (April to September 2014); there were 15 terms. The number of teachers targeted was 15, and the number of lecture subjects was 41. The number of responses was 1,678. Note that all the participants received the same questionnaire.

From the 1,678 comments, we manually evaluated the top 100 comments with the greatest number of words. The statistics of the comments consisted of an average 15.71 characters, a standard deviation of 14.6, and a maximum of 351 characters; there were more than 100 characters and 4 entries, and more than 50 letters and 24 entries.

A six-stage evaluation was manually performed on a section of the comments; these were rated as “Very bad (rank 1),” “Bad (rank 2),” “Quite bad (rank 3),” “Quite good (rank 4),” “Good (rank 5),” or “Very good (rank 6).” Then, we evaluated the remaining 1,678 unrated comments.

## III. Mutual Evaluation Method of Words and Comments

### A. Estimating the ratings of comments and specific words

The mutual evaluation method was used to estimate the ratings of comments and specific words in the comments. The outline is shown in Figure 1.

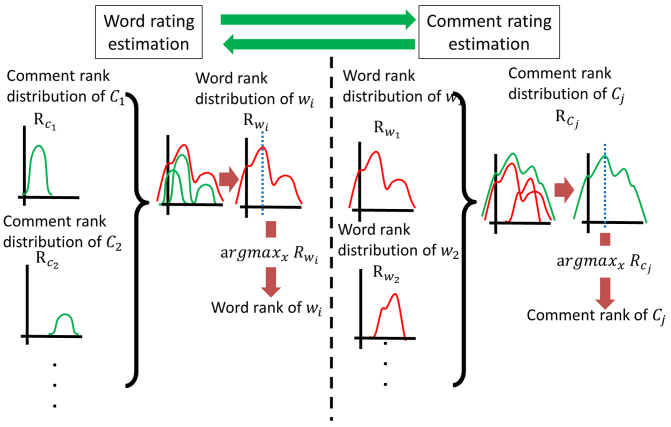


Fig. 1. Overview of mutual evaluation method of words and comments

(1) Word rating estimation: The ratings for words are estimated from the comments.

(1.1) Nouns, verbs, and adjectives are extracted from the comments, and the comment evaluation results evaluate the included words.

(1.2) For the evaluation rating, we created the comments including the target word $w_k$ as $i(= 1, \ldots, M)$, $\mu_i(= i)$ based around the evaluation $\mu_i(= i)$ for each expression of the estimated evaluation; we took the distribution $\sigma^2$ as the normal distribution $\phi(x; \mu_i, \sigma^2)$.

The number of answers for which the word $w_k$ can be the evaluation $i$ is $N_{w_k}(i)$. A word rank distribution was created by $\phi(x; \mu_i, \sigma^2)$; it was based on the estimated evaluation results frequency $N_{w_k}(i)$ for each word $w_k$.

(1.3) The normal distributions of each estimated evaluations were joined to create a mixed normal distribution, and this was the word rank distribution. By taking the mixed number (equal to the number of evaluations) in the mixed normal distribution as $M$ and taking the parameter $\alpha$ as the weight of the normal distribution in relation to the evaluation $i$, we defined the mixed normal distribution $p_{w_k}(x)$ in terms of the following formula. The initial value was set to $\sum_{i=1}^{M} \alpha_i = 1$.

$$p_{w_k}(x) = \sum_{i=1}^{M} \alpha_i \cdot \phi(x; \mu_i, \sigma^2) \cdot N_{w_k}(i)$$

(1-4) Let the maximum rank from the word rank distribution be the word rank estimate. Also, Figure 2 shows the word rank distribution byfor the mixed normal distribution using the word ”explanation(setumei)” as an example.

$$R_{w_k} = argmax_{i=1,\ldots,M}\, p_{w_k}(i)$$

Comment rating estimation: We created a comment rank distribution from the word rank distribution. Using this comment rank distribution, we set the comment rank estimated value to the maximum rank.

(2.1) To create the comment rank distribution $P_{c_l}$ for the comment $c_l$, it is necessary to consider the dependencies within the

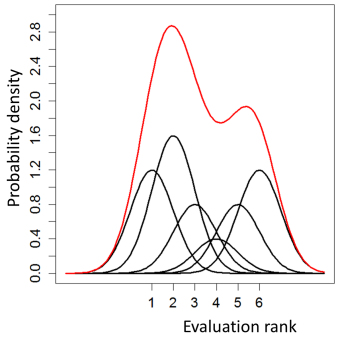


Fig. 2. Word rank distribution of “explanation”

| word \ Rank | Comment rank distribution | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| exercise 演習 | 1.18 | 1.22 | 1.18 | 1.14 | 1.15 | 1.14 |
| many 多い | 1.20 | 1.23 | 1.19 | 1.15 | 1.12 | 1.10 |
| fun 楽しい | 1.01 | 1.05 | 1.14 | 1.22 | 1.30 | 1.27 |
| exercise × many | 1.42 | 1.51 max | 1.40 | 1.31 | 1.29 | 1.25 |
| exercise × many × fun | 1.43 | 1.59 | 1.59 | 1.60 | 1.68 max | 1.59 |

Fig. 3. Process of comment rank distribution

comment. First, one adds to the probability pWk of the word rank distribution for the rank $i$ of the constituting words $w_k$. To reflect the dependency information, we multiplied the probabilities of the word estimations having dependencies these were included in the comments as $N_{c_l}(i) = \prod_{w_k \in c_l}(p_{w_k}(i)+1)$. The distribution $(N_{c_l}(1), N_{c_l}(2), \ldots, N_{c_l}(M))$ was made for each rank of comments. Next, the distribution for each rank of comments was approximated by a mixed normal distribution with the rank number as the mixture number using the EM algorithm[21]; the distribution was normalized so that the sum of the weights $\sum_{i=1}^{M} \beta_i = 1$ (see Figure 3) could be given as:

$$P_{c_l}(x) = \sum_{i=1}^{M} \beta_i \cdot \phi(x; \mu_i, \sigma^2) \cdot N_{c_l}(i)$$

(2-2) Similar to (1-4), the maximum rank was calculated from the comment rank distribution by using the comment rank estimate.

$$R_{c_l} = argmax_{i=1,\ldots,M}\, P_{c_l}(i)$$

(3) Estimation for all comments: We alternately repeated the comment estimation for all the comments and the word selection estimation for the words that comprise it; this was repeated until there were no further improvements to the estimated value for all comments. After the estimations stop repeating, the maximum selection from the comment rank distribution and word rank distribution become the final estimation value for the comments and the words.

(3.1) For all the comments, the comment rank distribution $P_{c_l}$ of the comment $c_l$ was updated as in (2.1).

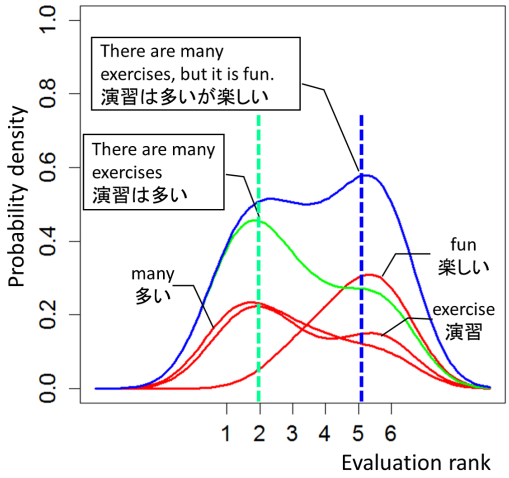


Fig. 4. Example of comment rank distribution

(3.2) Using a comment rank distribution in relation to all comments, we updated the word rank distribution of the words that constituted the comments. The word rank distribution $p_{w_k}(x)$ of the word $w_k$ was created by attaching a weight to the comment rank distribution $P_{c_l}$. The comment rank distribution $P_{c_l}$ of the comments $c_l$ to which the word $w_k$ belonged was summed up to create $\sum_{c_l \in W(w_k)} P_{c_l}(x)$. Similar to (2-1), we obtained the distribution $(N'_{w_k}(1), N'_{w_k}(2), \ldots, N'_{w_k}(M))$ for each rank. By using the EM algorithm, this was approximated by a mixed normal distribution with the rank number as the mixture number. Note that $W(w_k)$ expresses the comment group including the word $w_k$. Furthermore, this was expressed as a mixed normal distribution, and the total weight was normalized to $\sum_{i=1}^{M} \gamma_i = 1$. As an example, the comment rank distribution is shown in Figure 4 with the comment ”There are many exercises, but it is fun.”

$$p_{w_k}(x) = \sum_{i=1}^{M} \gamma_i \cdot \phi(x; \mu_i, \sigma^2) \cdot N'_{w_k}(i)$$

(4)Parameter estimation for comment rank distribution: When estimating the rating of each comment, it is necessary to estimate a rating with a small difference from the comment rating performed by humans. In this study, we estimated using an approximate solution with the steepest descent method so that the difference between weight $\alpha_i$ and normalized distribution dispersion $\sigma^2$ parameters with those ranks answered by the person concerned is minimized. The initial value was generated five times at random, and the value with the best approximate solution was used after applying the steepest descent method.

(5) Removing words by outliers of estimation: The comment analysis method proposed in this study, which includes multiple evaluators for seed comments, gives a higher polarity evaluation for words. Therefore, to address the evaluation skew for the evaluation estimation rank of words obtained from the

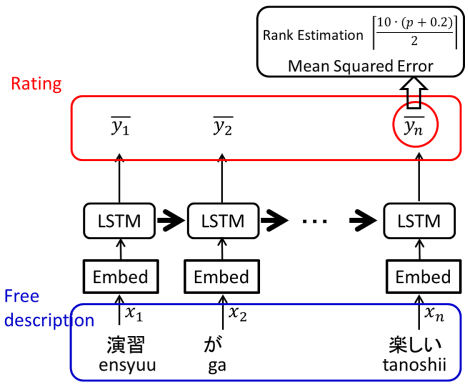


Fig. 5. Overview of estimation of rating by LSTM

seed comments, the outlier was sought through a one-sided test (5Here, the comments were evaluated after removing 47 words having a standard deviation of 1.14 or higher from the covered range.

## IV. Analysis of Free Description by LSTM

The comment data is divided into words by performing word segmentation. The evaluation rank is normalized from 0 to 1. The comment data and a pair of manual ranking evaluation ranks were used as the seed data for machine learning.

The evaluation environment using LSTM uses the Tensorflow version 1.5.0. The cell size of LSTM is 256; the number of layers is 1, and the embedded size of the word vector is 202. The number of occurrence words of the comment data is 1485 so that it is not reduced too much. The batch size was 10, and the initial learning rate was 0.001. The learning number (epoch number) was 10, and the loss rate was the mean square error. The loss rate does not change when the number of learning events is 11 or more; the number of times of learning was set to 10 times.

The configuration of LSTM is shown in Figure 5. The intermediate layer consists of the embedded layer and the LSTM layer; each block outputs the evaluation at that time to the output layer. Also, the output was taken as the input of the LSTM layer for the next time. Dropout was applied to the LSTM layer, and a sigmoid function was used as the activation function. In the output layer, the output of the last word of the comment was output as the estimation result of the evaluation label for the comment. For the loss function, the mean square error was used to calculate the loss rate from the difference between the output result and the human evaluation label; the parameters were then updated. The probability gradient method in the parameter update uses Adam. The number of epochs to repeat the learning is 10. The estimation process obtains the polarity values p from 0 to 1. The polarity value p is classified into six categories by the conversion function $\lceil \frac{10 \cdot (p+0.2)}{2} \rceil$.

In the closed test that manually evaluated comments by the mutual evaluation method and the LSTM evaluation method,

TABLE I
Closed test of comment evaluation (correlation between manually evaluation and estimated value)

| Evaluator | Mutual evaluation | LSTM |
|---|---|---|
| A | 0.773 | 0.864 |
| B | 0.482 | 0.813 |
| C | 0.359 | 0.894 |
| D | 0.573 | 0.813 |
| E | 0.521 | 0.846 |
| F | 0.475 | 0.850 |
| G | 0.284 | 0.841 |
| H | 0.734 | 0.907 |
| I | 0.779 | 0.756 |
| J | 0.535 | 0.877 |
| K | 0.657 | 0.889 |
| L | 0.661 | 0.841 |
| Average | 0.569 | 0.849 |

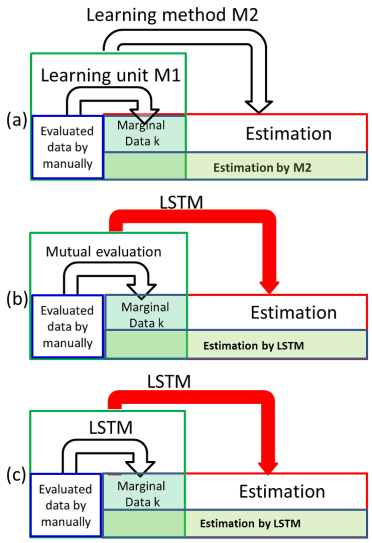


Fig. 6. Overview of estimation of rating by Bootstrap

the correlation coefficient was an average of 0.569 for the mutual evaluation method. The average by the LSTM evaluation method was 0.849. These results show that the LSTM evaluation method has better accuracy.

### A. Evaluation of the methods

The correlation coefficients between the evaluation values of the evaluator and the estimated values for 100 comments are shown in Table I. From the results of the closed test, we can see that the average accuracy for the machine learning using LSTM is 0.849 and the average accuracy for the comments and word mutual evaluation method is 0.569, which is a good result. In contrast, Table 2 shows the estimation result of 1542 unrated comments. The estimation results in many unrated comments show that the estimated average of machine learning using LSTMs is lower than the estimated average of the comments and words mutual evaluation method.

TABLE II
AVERAGE ESTIMATE OF RATINGS FOR UNRATED COMMENTS

| Comment | Mutual evaluation | LSTM | Imcorporating method($M_1$, $M_2$) | |
|---|---|---|---|---|
| | | | (Mutual evaluation,LSTM) $k = 100$ | (LSTM,LSTM) $k = 100$ |
| I think that it is easy to understand<br>わかりやすいと思います | 4.500 | 3.500 | 4.583 | 4.667 |
| I'd like writing on the blackboard.<br>板書がよい | 2.167 | 3.167 | 3.250 | 2.750 |
| Easy to understand lecture<br>授業が分かりやすい | 4.333 | 3.083 | 4.417 | 4.333 |
| Quick erase on the blackboard<br>黒板を消すのが速い | 2.333 | 3.000 | 3.333 | 3.167 |
| Learn how to make CG<br>CG の作り方を学べる | 3.333 | 3.000 | 3.500 | 3.583 |
| I understand how mathematics is actually used<br>数学が実際にどのように利用されているかがわかる | 4.500 | 1.750 | 4.333 | 4.417 |
| Because it is a practical subject, learn skills by exercises and tasks<br>実技教科なので，演習や課題で技術が身につく | 2.833 | 1.583 | 3.917 | 4.000 |
| I want you to do a firm check on the answer to the assignment<br>課題の答え合わせをしっかりやってほしい | 1.167 | 1.500 | 3.000 | 3.000 |
| The voice is small. Please let me know the calculation with numbers<br>声が小さい．数字を入れた計算を教えてほしい | 1.167 | 1.583 | 2.500 | 2.500 |
| Voice is not lost, I have difficulty, you do not check the students<br>声がとおっていない，ききづらい，生徒をみない | 1.583 | 1.000 | 1.417 | 1.667 |
| Average of estimation for unrated comments | 2.790 | 2.316 | 3.425 | 3.408 |

## V. COMBINING TWO METHODS BY INCORPORATING METHOD

The LSTM evaluation method tends to give a low estimation in the closed test. Therefore, we propose a method for incorporating $k$ of the newly evaluated comments; learning is done by merging the new learning data and repeating the next evaluation and learning. The procedure of the machine learning based on the learning-data incorporating method is described below. The overview is shown in Figure 6(a).

Step 1: For the initial learning, we learned the manual evaluation comments using the learning method $M_1$ and evaluated the unrated comments with the learning method $M_1$.
Step 2: We combined the newly evaluated $k$ comments into the learning data to create new learning data. If the $k$ newly evaluated data could be merged into the learning data, we moved to Step 3.
If $k$ newly evaluated data could not be merged into the learning data, the algorithm stops were performed.
Step 3: Here, the merged new learning data used the method $M_2$. The estimated unrated comments were not included in new learning data. After making an estimate, we returned to Step 2.

In this study, we examined two combinations of methods (Mutual evaluation method and LSTM) and (LSTM and LSTM) for the ($M_1$, $M_2$) learning methods. Each method is shown in Figure 6 (b) and (c). The total number of comments was 1678. There were 100 manual learning comments, and the number of data to be newly incorporated into the learning data was $k = 100$. For each of the two combinations, Table II shows the estimates of the unrated comments and the average of all the unrated comments for the two combinations.

We found some variations in each evaluation. The average values of all the unrated comments were 3.425 and 3.408 in the transfer method. When using the mutual evaluation method and the LSTM evaluation method alone, the estimated values were higher than the averages of 2.790 and 2.316.

## VI. CONCLUSION AND FUTURE WORKS

In this study, we have proposed a method for rating students' comments using the comments and words mutual evaluation method and the LSTM evaluation method. We have also proposed a method that combines the two methods by applying the bootstrap method to learn data. Regarding the evaluation of unrated comments, we think that the combination method improved the estimation of the unrated comments.

However, it is necessary to discuss the accuracy of the result for evaluating the unrated comments. Currently, the estimated values have been compared with the proposed method. We are considering a method to manually reevaluate the unrated comments. In our future studies, we will evaluate the estimation results. We will also evaluate students ' comments and words by using neural networks. In addition, we would like to consider a method for using neural networks inside the mutual evaluation method.

The method proposed in this study, and the rating item is set to the polarity of sentiment; however, there is no problem in changing the rating questionnaire item. Therefore, it is possible to evaluate the ratings with other evaluation questionnaire items. We think that we can analyze the word semantic by increasing the types of evaluation items used by words.

Also, we will consider the evaluation items of the sentences composed of words to be evaluated by a synthesis of words.

## REFERENCES


[1] J. W. Turban, "Students Prefer Audience Response System for Lecture Evaluation," in *Int. J. Emerging Technologies in Learning*, Vol. 6, No. 4, pp. 52–55, 2011.

[2] L. Jarahi, M. Najaf, "Evaluation of teaching through lecture with new methods of student-centered teaching in medical students," in *Future of Medical Education J.*, Vol. 3, No. 4, pp. 6–9, 2013.

[3] *Ministry of Education*, http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo4/003/gijiroku/06102415/004.htm

[4] M. Asahara, Y. Matsumoto, "Extended Models and Tools for High-performance Part-of-Speech Tagger," in *Proc. COLING 2000*, pp. 21–27, 2000.

[5] T. Kudo, Y. Matsumoto, "Japanese Dependency Analysis using Cascaded Chunking," in *CoNLL 2002: Proc. 6th Conf. Natural Language Learning 2002 (COLING 2002 Post-Conference Workshops)*, pp. 63–69, 2002.

[6] K. Nigam, et al., "Text Classification from Labeled and Unlabeled Documents using EM," in *Machine Learning*, Vol. 39, No. 2, pp. 103–134, 2000.

[7] C.D. Manning, H. Schutze. *Foundations of statistical natural language processing*, MIT press, 1999.

[8] P.D. Turney, "Thumbs up? thumbs down? Semantic Orientation Applied to Unsupervised Classification of Reviews," in *Proc. 40th Ann. Meeting Association for Computational Linguistics*, pp. 417–424, 2002.

[9] T. Inui, M. Okumura, "A Survey of Sentiment Analysis," *J. Natural Language Processing*, Vol. 13, No. 3, pp. 201–241, 2006.

[10] I. Rish, "An empirical study of the Naive Bayes classifier," in *IJCA 2001 Workshop Empirical Methods Artificial Intell.*, Vol. 3, No. 22, pp. 41–46, 2001.

[11] V. Vapnik, *Statistical Learning Theory*, Wiley, 1998.

[12] J. Shawe-Taylor, N. Cristianini, *Kernel methods for pattern analysis*, Cambridge University Press, 2004.

[13] H. Takamura, T. Inui, M. Okumura, "Extracting Semantic Orientations Using Spin Model," *IPSJ Vo.47*, No. 2, pp. 627–637, 2006. (In Japanese)

[14] H. Takamura, *Semantic Orientations of Words*, [Online]. Available: http://www.lr.pi.titech.ac.jp/~ takamura/pubs/pn_ja.dic

[15] Greff, K. et al., "LSTM:A Search Space Odyssey," *IEEE Trans. Nenural Netw. Learn. Syst.*, Vol. 28, Issue. 10, pp. 2222-2232, 2017.

[16] M. Wollmer, et al., "YouTube Movie Reviews: Sentiment Analysis in an Audio-Visual Context," *IEEE Intell. Syst.*, Vol. 28, Issue. 3, pp. 46–53, 2013.

[17] L.Zhang, S. Wang, B. Liu, "Deep Learning for Sentiment Analysis: A Survey," arXiv preprint arXiv:1801.07883v2. 2018.

[18] K. Nakanishi, N. Kobayashi, H.Shiina, F.Kitagawa, "Estimating word difficulty using semantic descriptions in dictionaries and Web data," *Proc. of 2012 IIAI International Conference on Advanced Applied Informatics (IIAIAAI 2012)*, pp. 324–329, 2012.

[19] P.Q. Zang, H. Shiina, N.Kobayashi, F.Kitagawa, "Chinese and Japanese Word Learning System by Estimation of Word Difficulty," *Proc. of ACIS 2012, The second asian conference on information systems*, pp. 320–323, 2013.

[20] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[21] A.P. Dempster, N. M. Laird, D.B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society*. Series B, No. 39, Vol. 1, pp. 1–38, 1977.