Sentiment Analysis on the Online Reviews Based on Hidden Markov Model

Xiaoyi Zhao and Yukio Ohsawa
Department of Systems Innovation, School of Engineering, The University of Tokyo, Tokyo, Japan
Email: issyou5072@g.ecc.u-tokyo.ac.jp, ohsawa@sys.t.u-tokyo.ac.jp

***Abstract*—In this study, a new sentiment analysis model of online-shopping reviews based on hidden Markov model has been proposed. Both the influence of the latest two comments and the most popular comment from the Amazon Japan review page are taken into consideration. The supervised training method is used to train this model, and then the model is optimized by using a variation of genetic algorithm. The performance is evaluated through an experiment of sentiment classification of online-shopping reviews of Amazon Japan's tea category comparing to other methods from previous ones such as Support Vector Machine, Logistic Regression with built-in cross-validation and so on. The result shows that the adapted hidden Markov model has the highest f1 score among the other baseline methods.**

***Index Terms*—HMM, 2 dimensional HMM, machine learning, sentiment analysis, online review**

I. Introduction

With the development of the Internet, people are used to online shopping. After these online shopping behaviors, consumers are likely to leave their precious comments on the webpage, to express their sentiments about the shopping experience. For the owners of the online shops, they can improve their products and service through reviews. On the other hand, the online shopping reviews are most helpful for the latent consumers to decide whether to buy the product or not.

Sentiment analysis is a method to recognize the sentiment behind a text, a voice clip and so on. It can be applied to help the understanding of the sentiment behind the online shopping reviews. There are two directions in the sentiment analysis field. The first direction is exerting dictionaries to analyze text sentiment. The most important feature of dictionary-based text sentiment analysis is that the dictionary contains the word of the analyzed article. This feature can help dictionary-based text sentiment analysis to quickly and accurately recognize the text sentiment. This feature, however, is also the drawback of this direction. Some internet words constantly used in the review text, such as "LOL", "w"[1] [1] are not included in the dictionary due to their fast change rate. Therefore, those types of new internet words cannot be precisely recognized.

Another direction of sentiment analysis field is using machine learning. Generally, the bigger the training data set, the higher the accuracy of the result. Nonetheless, in most review sites of commercial products, each product has limited comments, putting a constraint on the size of the data set. Methods like Neural Networks (NNs) cannot be conducted due to the small data size. Meanwhile, because of the complexity of neutral network structure, explaining the trained NN models has become a difficult task. Imaging the situation that NN classifies a patient to the class who already have cancer, but the reason why NN classifies that patient to the cancer class has not yet been explained completely. It can then be found that the accurate classification is not helpful, since a result whether the patient has the cancer or not cannot help the doctor to decide the therapeutic method. Although NNs can conduct notable achievements in some tasks, lack of explainablility makes the result untrustworthy depending on certain cases.

II. Literature Review

As has been mentioned in the introduction, the sentiment analysis has two directions, which are dictionary-based field and machine learning field. In the direction of dictionary-based, the essential process starts by dividing the text into single words, and then building a dictionary containing the information of dependency relationship, text construction and sentimental words. Finally, the sentiment score is calculated. For instance, the study of [2] uses about 1,7000 pieces of news to compute the sentiment score. Similarly, the method of [3] also uses news data to calculate the sentiment score while the influence of adverbs of degree, negation or affirmation of the sentence and tenses of sentences are taken into consideration to compute the sentiment score of sentence.

Using machine learning method is another direction of sentiment analysis. In this direction, the typical process is to first divide the text into single words, and uses some methods converting the words to the word vectors such as tf-idf, word2vec and so on. Then the main procedure is to train the machine learning model and eventually use the trained model to classify the text to each sentiment category. For instance, [4] uses a model comprising of both unsupervised and supervised method to learn word vectors, and thus able to catch semantic term document information and recognize sentiment content. The paper

Manuscript received November 4, 2017; revised May 3, 2018.
[1] "w" means laugh in Japan, because it looks like a smiley mouth.

doi: 10.12720/jait.9.2.33-38

of [5] is an empirical study looking into the text-based sentiment prediction problem supported by supervised machine learning.

With the boom of deep learning, some sentiment analysis method has emerged using deep leaning to improve the accuracy of classification. The paper [6] is a typical representative of this method using deep learning techniques to overcome the limits that many feature selection methods focuses on only linear relationships between features. Kim, Lee and Provost use multimodal data to capture complex non-linear feature interactions. Meanwhile, some creative methods which neither belong to the direction of dictionary-based nor the direction of machine learning are applied to sentiment analysis. As a representative paper of this field, [7] takes a simple web co-occurrence approach combining the frequency count information of search engines to calculate sentiment score.

In this paper, after the essential steps of sentiment analysis in machine learning field, an adapted hidden Markov model is proposed. Since one of the features of reviews is that it contains the sentiment of customers about the product or service they have spent money on. Thus, comment writers should have some sentiment al sentiments about the product or service. From this perspective, the hidden Markov model fits the goal of this paper.

III. Hidden Markov Model

Hidden Markov model (HMM) is a statistical model in which the dynamic system being modeled is assumed to be a Markov process with unobserved states [8]. HMM uses three parameters to describe the relationship between observation series and unobservation (hidden) series, which are called transition matrix ($\mathbf{A}$), emission matrix ($\mathbf{B}$) and start matrix (π). Fig. 1 shows a simple HMM. Using three parameters, HMM can generate the maximized posterior probability of hidden series from observation series.

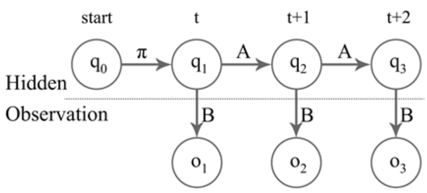

Figure 1. A simple Hidden Markov model.

Sentiment analysis using HMM is usually 1 dimensional HMM (1dHMM), which is the classical HMM. 1dHMM is the model that present hidden status only influenced by previous hidden states. Similarly, the observation states are the output of the present hidden states. 1dHMMs are widely used in the fields such as HMM-based speech sentiment recognition [9] and speaker characteristics recognition [10]. Nevertheless, research about analysis on reviews using HMM is relatively limited.

Parameters of HMM are defined as follows:

$$\mathbf{A} = \left[a_{ij}\right]_{N\times N} \tag{1}$$

$$a_{ij} = P(I_{t+1} = q_j | I_t = q_i), \quad i = 1,2,\cdots,N; \quad j = 1,2,\cdots,N \tag{2}$$

a_{ij} is an element from the matrix $\mathbf{A}$ showing the probability that at time $t+1$, the hidden state is $I_{t+1} = q_j$, under the condition of hidden state $I_t = q_j$. N is the number of categories of sentiments.

$$\mathbf{B} = \left[b_j(l)\right]_{N\times M} \tag{3}$$

$$b_j(l) = P(O_t = o_l | I_t = q_j), \quad l = 1,2,\cdots,M; \quad j = 1,2,\cdots N \tag{4}$$

$b_j(l)$ represents the probability that under the condition of hidden state $I_t = q_j$, the observation phenomenon is $O_t = v_l$. Whereas M means the number of observation states.

$$\pi = [\pi_i]_{N\times 1} \tag{5}$$

$$\pi_i = P(I_1 = q_i), \quad i = 1,2,\cdots,N \tag{6}$$

π_i refers to the probability of hidden state $I = q_i$ when the time is $t = 1$.

Because the transition of the hidden states is only influenced by the previous hidden states, the 1dHMM model generates relative simple model. However, in reality, the transition of hidden states is more complex. Therefore, a transition matrix to flexibly describe the transitional relationships is required.

Figure 2. Amazon Japan review page layout. When customers click the "write a customer review" button, they will be unconsciously influenced by top customer reviews and most recent customer reviews.

IV. 2 Dimensional Hidden Markov Model

2 dimensional HMM (2dHMM) is an expanded HMM, in which the current unseen state is generated by states of two times before the current hidden state, and the observation state is an output of the current unseen state. In this study, data from Amazon Japan is used and its webpage layout has an impact on potential shoppers' behaviors. From human reading webpage behaviors [11] and the Amazon Japan review page layout, which is shown as Fig. 2, it is assumed that when customers click the button to write a customer review, they will unconsciously read the top and the latest two comments.

Thus, 2dHMM have been adapted accordingly. This study assumed that observation state is not only an output of current and previous unseen state but also the top rated unseen state. Fig. 3 illustrates this process. A higher dimension HMM can be applied to commercial Web pages in general, because customers view various information such as advertisements, list of products, and link to other services.

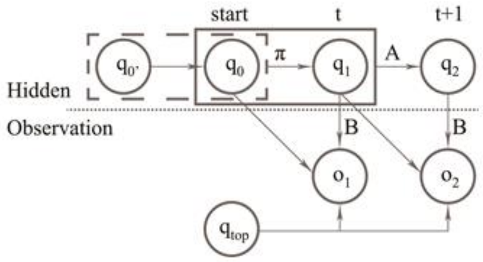

Figure 3. 2dHMM which considers the influence of the top comment and the latest two comments. In the hidden layer, each circle represents review's cluster number according to the clusters obtained by the review vectors. In the Observation layer, each circle represents the review's sentiment label. The order of reviews reflects the time stream $t, t+1, \cdots$

Flow chart of this 2dHMM model shows the process of this model how to work. Starting with the training stage, the initial parameter matrices are smoothened. The parameter matrices then enter the optimization stage, where several other steps are performed as detailed in Fig. 4.

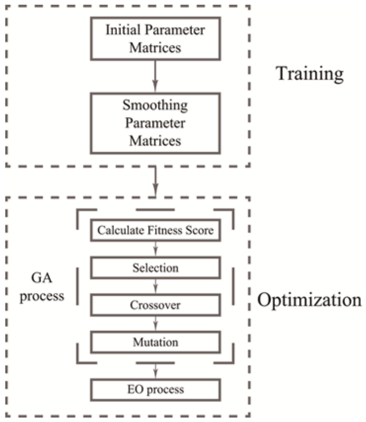

Figure 4. Flow chart of the adapted 2dHMM.

A. Initial Parameter Matrices

Similar to 1dHMM, the 2dHMM also has three parameters $\mathbf{A}$, $\mathbf{B}$ and π. The definitions are as follows:

$$\mathbf{A} = \left[a_{ijk}\right]_{N\times N\times N} \tag{7}$$

$$a_{ijk} = P(I_{t+2} = q_k | I_t = q_i, I_{t+1} = q_j), \quad (i,j,k) \in 1,2,\cdots,N \tag{8}$$

a_{ijk} is an element from the matrix $\mathbf{A}$ showing the probability that at time $t+2$, the hidden state is $I_{t+2} = q_k$, under the condition of hidden state $I_t = q_i$ and $I_{t+1} = q_j$.

$$\mathbf{B} = \left[b_{jk}(l)\right]_{N\times N\times M} \tag{9}$$

$$\begin{aligned} b_{jk}(l) &= \alpha P(O_{t+2} = o_l | I_{top} = q_{top}) \\ &+ (1-\alpha) P(O_{t+2} = o_l | I_{t+1} = q_j, I_{t+2} = q_k) \\ &\alpha \in [0,1); \quad (i,j) \in 1,2,\cdots N; \\ &l \in 1,2,\cdots,N \end{aligned} \tag{10}$$

$b_{jk}(l)$ represents the emission probability generated by the latest two hidden states and influenced by the top comment. α is the factor influencing the top comment. A simple reason why α cannot take 1 is that when α takes 1 the structure of this adapted 2dHMM does not follow the Markov process.

$$\pi = \left[\pi_{ij}\right]_{N\times N} \tag{11}$$

$$\pi_{ij} = P(I_1 = q_i, I_2 = q_j), \quad (i,j) \in 1,2,\cdots,N \tag{12}$$

The start probability π_{ij} refers to the probability of the unseen state series started by q_i and q_j.

Since the parameters of 2dHMM are difficult to directly obtain, and due to the low efficiency of Baum-Welch algorithm, this study uses supervised training method to obtain the parameters $\mathbf{A}$, $\mathbf{B}$ and π. The parameters $\mathbf{A}$, $\mathbf{B}$ and π are basically (i.e., before introducing the smoothing in *B* below) obtained by replacing P in Eq. (8), (10), and (12) with $\hat{P}$ in Eq. (13) and (14).

$$\hat{P}(*) = \frac{Count(*)}{n} \tag{13}$$

$$\hat{P}(*|\circ) = \frac{Count(*,\circ)}{Count(\circ)} \tag{14}$$

where $Count(*)$ means the number of "*" and n is the length of input series.

B. Smoothing Parameter Matrices

According to the training data set, some gaps may appear in the parameters matrices $\mathbf{A}$, $\mathbf{B}$ and π. These gaps may cause unpredictable problems. As the result, smoothing factor λ is added to smooth the matrix. After this process, the robustness of 2dHMM is improved. After smoothing, parameter matrices $\mathbf{A}$, $\mathbf{B}$ and π are rewritten as follows:

$$\begin{aligned} \mathbf{A}: \ \bar{P}(I_{t+2}|I_t, I_{t+1}) &= \\ &\lambda_1^A \hat{P}(I_{t+2}) + \lambda_2^A \hat{P}(I_{t+2}|I_{t+2}) \\ &+ \lambda_3^A \hat{P}(I_{t+2}|I_t, I_{t+1}) \end{aligned} \tag{15}$$

$$\begin{aligned} \mathbf{B}: \ \bar{P}(O_{t+2}|I_{t+1}, I_{t+2}) &= \\ &\lambda_1^B \hat{P}(O_{t+2}) + \lambda_2^B \hat{P}(O_{t+2}|I_{t+1}) \\ &+ \lambda_3^B \{\alpha \hat{P}(O_{t+2}|I_{top}) \\ &+ (1-\alpha) \hat{P}(O_{t+2}|I_{t+1}, I_{t+2})\} \end{aligned} \tag{16}$$

$$\pi: \ \bar{P}(I_1, I_2) = \lambda_1^{\pi}\hat{P}(I_1) + \lambda_2^{\pi}\hat{P}(I_1, I_2) \quad (17)$$

This smoothing method follow Brants' method [12] to calculate smoothing factors λ_i^A, λ_i^B, λ_j^{π}, where $i = 1,2,3$, $j = 1,2$ and $\sum_{i=1}^{3}\lambda_i^A = \sum_{i=1}^{3}\lambda_i^B = \sum_{j=1}^{2}\lambda_j^{\pi} = 1$.

C. Optimization

After the process *A* and *B*, a variation of genetic algorithm called GA-EO [13] is used to optimize the structure of 2dHMM. Due to the high local searching ability, this method allows 2dHMM to find optimized parameters around parameters $\mathbf{A}$, $\mathbf{B}$ and π in the post-training process.

i. GA process

a) Fitness function

This study uses the fitness function defined by Won, Prügel-Bennett and Krogh [14], which is shown as follows, to evaluate each individual's fitness score.

b) Selection

This study uses the roulette wheel selection method to select individuals. The better the individuals are, the more chances to be selected they have. This method gives weaker individuals a chance that they may survive the selection process and prove their useful of some component by following the crossover processing.

c) Crossover

This study uses the single-point crossover method to generate new individuals. The crossover point will be randomly chosen.

d) Mutation

This study uses boundary method to replace the genome with either lower or upper bound randomly. The change position will be randomly decided. The mutation scale will be randomly selected from 1% to 5% length of genome.

ii. EO process

After GA process [13] above, individuals are ranked according to the fitness function. The smallest fitness individual X, which has a fitness score of F_s, will be chosen. Dividing individual X's genome into p slices, X_i represents the i-th slice of the individual X. A new individual X_i^{new} will be generated after randomly changing X_i. The new individual X_i^{new} will then be rated by the fitness function, and the fitness score is marked as F_i. If $F_i < F_s$, the change of X_i is canceled. Otherwise, the change of X_i will be preserved. The EO process will enhance GA local optimized ability.

V. Experiment

For the purpose of this paper, all of the reviews of tea from Amazon Japan until 13th June, 2017 have been used. Products with more than 20 reviews are kept and this study finally collected 13213 reviews of 160 products in total.

The experiment started by using MeCab [15] as the word segmentation system, and followed by removing all of the numerals, auxiliary words, auxiliary verbs, punctuation characters and stop words [16], [17]. After word segmentation, the Word2Vec [18] model is trained by using the corpus of review words and Japanese Wikipedia article corpus. Using this Word2Vec model, an output vector is generated according to the input word. For each review, a text vector can be obtained by adding up all the word vectors and dividing by the number of words. This process is shown as follows:

$$n^{(i)} = \text{the number of words in the } i\text{-th review} \quad (18)$$

$$R_{word}^{(i)} = \{w_{1^{(i)}}, w_{2^{(i)}}, \cdots, w_{n^{(i)}}\}, w_s \xrightarrow{Word2Vec} v_s \quad (19)$$

$$R_{vector}^{(i)} = \{v_{1^{(i)}}, v_{2^{(i)}}, \cdots, v_{n^{(i)}}\}, \quad (20)$$

$$V_R^{(i)} = \frac{1}{n}\sum_{s=1}^{n} v_{s^{(i)}}, \quad s = 1,2,\cdots n \quad (21)$$

Then, the sentimental word vectors are obtained from Word2Vec model's outputs of four sentimental tags[2].

Next, the K-Means [19] clustering method is used to portion reviews' text vector into 55 clusters. The index of each cluster is used as observation states.

Then, the reviews' order is randomly shuffled and four sentimental tags are labeled through online surveys in the format of multiple choice[3]. Since one survey is answered by multiple participants and consider the different understandings of each review by different participants, the most voted answer is used as the baseline answer. For each review, the baseline answer reflects the sentiment of the comment writer. These baseline answers are then used as hidden states.

The experiment then proceeds by dividing the data into training data and test data in the ratio of 8:2. The following methods are used as baselines to compare with 2dHMM.

i. Random

In this method, the sentiment category numbers are generated randomly. Then it is tested by the true sentiment categories.

ii. 1dHMM

Similar to 2dHMM, the index of each cluster which is used by 2dHMM are used as observation states. And sentiment category numbers are used as hidden states.

iii. Cosine similarity (CS)

In this method, the cosine score is calculated between each text review vector and the four sentimental vectors. The sentimental word vector which gives the highest cosine score is selected and its corresponding sentimental category is marked as the sentiment of that text review vector.

iv. ML-Ask

ML-Ask is a python tool which is based on a linguistic assumption that sentimental states of a speaker are conveyed by sentimental expressions

[2] Four sentimental tags: joy, anger, sadness, anticipation

[3] Both one answer and more than one answer are accepted.

used in emotive utterances [20]. The test review's text is used as input, the output of ML-Ask is used as the category of the sentiments

v. *Support Vector Machine (SVM)*

In this method, Reviews' text vectors are used as input vectors and the corresponding category of sentiments are the outputs.

vi. *Logistic Regression with built-in cross-validation (LRCV)*

Similar to the Cosine similarity method, the test review vectors are used as input vectors. The output is the category of sentiments.

There are four types of sentiments to be classified, thus this study has to adapt the rating method from the original non-weighted macro-average precision, recall and f1 score. Starting by calculating the macro-average precision, recall and f1 score, they are then weighted by support, which refers to the number of true instances under each sentiment label. These changes "macro" to account for label imbalance. It can result in an f1 score that is not between precision and recall. By using this rate method, the level of bias, which appears due to the different experiment methods, can be reduced.

The results of 2dHMM and other base methods are shown as in Table I.

TABLE I. RESULTS COMPARE WITH OTHER METHODS (UNDER WEIGHTED AVERAGE LEVEL)

Method	Precision	Recall	F1
Random	0.4369	0.2605	0.2992
1dHMM	0.4903	0.5526	0.4430
CS	0.3871	0.1758	0.1782
ML-Ask	0.4489	0.4615	0.4450
SVM	0.3575	0.5980	0.4475
LRCV	0.4658	**0.6176**	0.4874
2dHMM	**0.5496**	0.6039	**0.5379**

From the table, in terms of recall, the best result appears in LRCV method. Nevertheless, 2dHMM also shows a high recall very close to the best result comparing with the other methods. Furthermore, 2dHMM has the highest precision score than the rest of the methods in the table. Overall, 2dHMM also have the highest f1 score among the methods in comparison.

VI. CONCLUSIONS

While a lot of different methods have been used for sentiment analysis for text from past studies, this study has chosen the Hidden Markov model. By this device, it is possible to take consideration of the webpage layout from Amazon Japan's product review page under the human webpage reading behaviors. The analysis of this paper creatively takes the latest two reviews and top-rated review from Amazon Japan's tea category into consideration. Since more influential factors than 1dHMM and other base line methods are considered, 2dHMM has shown the highest precision and f1 score. Such an extensive design of analysis model to higher dimension HMM is applicable not only to Amazon Japan, but also more to general commercial Web pages. This is because customers view multi-fold information in general from a Web page, as far as the page is designed to include complex information such as advertisements, list of products, and links to other services. In the future studies, the plan is to adapt our model from batch learning to online learning to better accomplish the goal of sentiment analysis for online reviews.

ACKNOWLEDGMENT

This work was supported by JST CREST Grant Number JPMJCR1304, JSPS KAKENHI Grant Numbers JP16H01836, and JP16K12428.

REFERENCES

[1] M. Iwasaki, R. Maeda, and H. Kawashima, "Expressions of the internet that young people pay attention to internet slang and dialect (in Japanese)," *The Bulletin of Hachinohe Institute of Technology*, vol. 36, pp. 41–56, 2017.

[2] T. Kumamoto and K. Tanaka, "Extracting feelings from newspaper accounts on the web (in Japanese)," *IPSJ Natural Language Processing*, vol. 2005, no. 1 (2004-NL-165), pp. 15–20, 2005.

[3] H. Sugawara, A. Neviarouskaya, and M. Ishizuka, "Affect extraction from Japanese text using emotional dictionary (in Japanese)," in *Proc. 23rd Annual Conference of the Japanese Society for Artificial Intelligence*, 2009, pp. 1–2.

[4] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proc. 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011, pp. 142–150.

[5] C. O. Alm, D. Roth, and R. Sproat, "Emotions from text: Machine learning for text-based emotion prediction," in *Proc. Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, 2005, pp. 579–586.

[6] Y. Kim, H. Lee, and E. M. Provost, "Deep learning for robust feature generation in audiovisual emotion recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 3687–3691.

[7] Z. Kozareva, B. Navarro, S. V ́azquez, and A. Montoyo, "Ua-zbsa: a headline emotion classification through web information," in *Proc. 4th International Workshop on Semantic Evaluations*, 2007, pp. 334–337.

[8] S. R. Eddy, "Hidden markov models," *Current Opinion in Structural Biology*, vol. 6, no. 3, pp. 361–365, 1996.

[9] A. Nogueiras, A. Moreno, A. Bonafonte, and J. B. Marino, "Speech emotion recognition using hidden markov models," in *Proc. Seventh European Conference on Speech Communication and Technology*, 2001.

[10] Y. Jin, T. Sakuma, S. Kato, and T. Kunitachi, "Estimating personality impression from speech record using hidden markov models (in Japanese)," *IEEJ Transactions on Electronics, Information and Systems*, vol. 135, no. 12, pp. 1517–1523, 2015.

[11] L. Lorigo, *et al.*, "Eye tracking and online search: Lessons learned and challenges ahead," *Journal of the Association for Information Science and Technology*, vol. 59, no. 7, pp. 1041–1052, 2008.

[12] T. Brants, "Tnt: A statistical part-of-speech tagger," in *Proc. Sixth Conference on Applied Natural Language Processing*, 2000, pp. 224–231.

[13] J. He and X. D. Li, "Improved genetic algorithm: Ga-eo algorithm (in Chinese)," *Application Research of Computers*, vol. 29, no. 9, pp. 3307–3311, 2012.

[14] K. J. Won, A. Prugel-Bennett, and A. Krogh, "Training hmm structure with genetic algorithm for biological sequence analysis," *Bioinformatics*, vol. 20, no. 18, pp. 3613–3619, 2004.

[15] T. Kudo. Mecab: Yet another part-of-speech and morphological analyzer. [Online]. Available: http://ci.nii.ac.jp/naid/10015541516/

[16] English stopword list. (October 2017). [Online]. Available: http://svn.sourceforge.jp/svnroot/slothlib/CSharp/Version1/SlothLib/NLP /Filter/StopWord/word/English.txt
[17] Japanese stopword list. (October 2017). [Online]. Available: http://svn.sourceforge.jp/svnroot/slothlib/CSharp/Version1/SlothLib/NLP /Filter/StopWord/word/Japanese.txt
[18] R. Rehurek and P. Sojka, "Software framework for topic modelling with large corpora," in *Proc. LREC Workshop on New Challenges for NLP Frameworks*, Valletta, Malta, May 2010, pp. 45–50.
[19] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.
[20] M. E. Ptaszynski, "Ml-ask". (October 2017). [Online]. Available: http://arakilab.media.eng.hokudai.ac.jp/ptaszynski/repository/mlask.htm

Xiaoyi Zhao was born in Chengdu, Sichuan Province, China on November 13 1991. He graduated from Chengdu College of University of Electronic Science and Technology of China located in Chengdu, Sichuan Province, China with bachelor of computer science in June 2014. Presently he is a master student from department of System Innovation, School of Engineering of the University of Tokyo in Tokyo, Japan.

Yukio Ohsawa was born in Kyoto, Japan, April 1st 1968, and obtained BE (1990), ME (1992), and PhD (1995) in electronic engineering, The University of Tokyo, Tokyo, Japan.

He worked for the School of Engineering Science in Osaka University (Research Associate, 19951999), School of Business Sciences in University of Tsukuba (Associate Professor, 1999-2005), then moved back to the School of Engineering, The Univ. of Tokyo, Tokyo, Japan (Professor since 2009). He created a new domain called chance discovery, meaning to discover events of significant impact on decision making, since year 2000. His original concepts and technologies have been published as books such as "Chance Discovery" (2003 Springer), "Innovators' Marketplace: Using Games to Activate and Train Innovators" (2012 Springer). He edited special issues as guest editors for journals, mainly relevant to chance discovery, such as Information Sciences (2009), New Generation Computing (2003), Journal of Contingencies and Crisis Management (2002), etc. He also published 100 journal papers, and more presentations in conferences including International Joint Conference on Artificial Intelligence (IJCAI), World Wide Wed (WWW), Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), etc.

Prof. Ohsawa is a member of IEEE, Japanese Society of AI, and also contributed to program committees of conferences including IJCAI, the editorial board of a number of interdisciplinary journals, and worked as the co-chair of IEA/AIE, and initiated the Technical Committee of Information Systems for Design & Marketing in of IEEE-SMC in 2005. His current main interest in the design of data, via the communication of data scientists, data providers, and users of data.