On robustness of the supervised multiclass classifier for autocoding system

**Keywords:** Coding, Machine learning, Multiclass classification

# Introduction

We developed a supervised multiclass classifier for autocoding based on reliability scores [1]. The purpose of this paper is to investigate the robustness of this classifier in coding tasks in official statistics.

Text response fields such as fields for occupation, industry, and household income and expenditure, are sometimes found on survey forms in official statistics. Those responded text descriptions are usually translated into corresponding classification codes for efficient data processing. Although, originally, coding tasks are performed manually, the importance of automated coding is increasing with the improvement of computer technology in recent years. Therefore, studies focused on developing an algorithm for autocoding have been seen in official statistics. For example, Hacking and Willenborg [2] introduced coding methods including autocoding techniques. Gweon et al. [3] illustrated methods for automated occupation coding based on statistical learning.

We also developed a supervised multiclass classifier for the coding task of *the Family Income and Expenditure Survey* in Japan. Originally, our classifier was developed based on a simple machine learning technique, and it performs exclusive classification [4], [5], [6]. However, the classifier incorrectly assigns classification codes for some text descriptions with ambiguous information because of the semantic problem, interpretation problem, and insufficiently detailed input information. The main reason for these problems is the unrealistic restriction that one text description is classified to a single code, we developed a new classifier [7] that allows for the assignment of one text description to multiple classification codes based on partition coefficient or partition entropy [8] and developed the further newly algorithm with a calculation of a new reliability scores [1]. In this proposed classifier based on the reliability scores, although we improved the classification accuracy, we need to consider not only the accuracy but the robustness of the classification. Generally, a classifier for the autocoding system requires robustness for the stable code assignment, whereas the style of text description is not always stable even in the same survey as it depends on respondents. Therefore, this study investigates the robustness of our classifier based on reliability scores with a numerical example using the noise-added survey data.

# Multiclass classifier for autocoding based on reliability scores

## Overview of the classifier

The classifier consists of training and classification processes. In the training process, the classifier performs feature extraction and creation of a feature frequency table. First, the classifier performs tokenizing each text description into words using MeCab [9], which is a dictionary-attached morphological analyzer. Then the classifier takes word-level N-grams from the tokenized word sequences as features. Finally, it tabulates all extracted features along with the given classification codes into a feature frequency table. In the classification process, our classifier performs feature extraction, the retrieval of candidate classification codes, and classification codes assignment. First, the classifier extracts features of target text descriptions. Then, it retrieves the corresponding classification codes from the feature frequency table provided by using the extracted features. After that, the classifier calculates the probability of *j*-th feature,$ j=1, …J$$j (j=1, …J)$ to a code$ k, k=1, …K$ that is defined as

$$p\_{ik}=n\_{ik}/n\_{j}, n\_{j}=\sum\_{k=1}^{K}n\_{jk} , (1)$$

where $n\_{jk}$ is the number of features in a code $k$ with a feature *j* in the training dataset. Then, we arrange $\left\{p\_{j1},\cdots ,p\_{jK}\right\}$ in descending order and create $\left\{\tilde{p}\_{j1},\cdots ,\tilde{p}\_{jK}\right\}$, such as $\tilde{p}\_{j1}\geq \cdots \geq \tilde{p}\_{jK}, j=1,\cdots ,J$. Next, we select at most $\tilde{K}, (\tilde{K}\leq K)$ promising candidate codes for each feature based on the values of $\tilde{p}\_{jk}$. That is, we create $\left\{\tilde{\tilde{p}}\_{j1},\cdots ,\tilde{\tilde{p}}\_{j\tilde{K}\_{j}}\right\}, \tilde{K}\_{j}\leq \tilde{K}\leq K$. In the case when we cannot select different $\tilde{K}$ codes, that is the case when there are same values in $\left\{\tilde{p}\_{j1},\cdots ,\tilde{p}\_{jK}\right\}$, then we select as many as possible different $\tilde{K}\_{j}$ codes for each feature *j*. Then, we define the reliability score $\overbar{p}\_{jk}$ utilized the partition entropy [9] as follows [1]:

$$\overbar{p}\_{jk}= \tilde{\tilde{p}}\_{jk}\left(1+\sum\_{m=1}^{\tilde{K}\_{j}}\tilde{\tilde{p}}\_{jm}log\_{K}\tilde{\tilde{p}}\_{jm}\right), j=1,\cdots ,J, k=1,\cdots ,\tilde{K}\_{j}. (2)$$

When the number of target text descriptions is *T*, and each text description includes $h\_{l}, l=1,\cdots ,T$ features, corresponding $\overbar{p}\_{jk}$ shown in (2) for *l*-th text description can be represented as

 $\overbar{p}\_{j\_{l}k}, j\_{l}=1,\cdots ,h\_{l}, k=1,\cdots ,\tilde{K}\_{j\_{l}} , l=1,\cdots ,T (3)$

which shows a reliability score of *j*-th feature included in *l*-th text description to a code *k*. The total number of the promising candidate codes for *l*-th text description is $\sum\_{j\_{l}=1}^{h\_{l}}\tilde{K}\_{j\_{l}}$. Then, the classifier selects top $L\in \{1,…,\sum\_{j\_{l}=1}^{h\_{l}}\tilde{K}\_{j\_{l}}\}$ codes for assignment of *l*-th text description based on the reliability score $\overbar{p}\_{j\_{l}k}$ shown in (3).

## Investigation of robustness

For investigating the robustness of the classifier [1] described in above section 2.1, we perform the following procedures.

(Step 1) Extract features, calculate $p\_{jk}$ and $\overbar{p}\_{j\_{l}k}$ shown in (1) and (3). Set $\overbar{p}\_{j\_{l}k}^{(0)}= \overbar{p}\_{j\_{l}k}$ .

(Step 2) Generate normal random numbers as

$$∆=\left(δ\_{jk}\right), j=1,… J, k=1,… K, δ\_{jk}\~ N\left(0, σ^{2}\right). (4)$$

(Step 3) Calculate $\hat{p}\_{jk}≡ p\_{jk}+δ\_{jk}$. Determine the promising candidates for each feature based on calculated $\hat{p}\_{jk}$ .

(Step 4) Calculate reliability scores, $\overbar{p}\_{j\_{l}k}$ by using (3) based on $\hat{p}\_{jk}$.

(Step 5) Determine top $L\in \{1,…,\sum\_{j\_{l}=1}^{h\_{l}}\tilde{K}\_{j\_{l}}\}$$L (L=1,…, \tilde{K})$$L (L=1, 2,3…)$$L (L=1, 2,3…)$ codes based on the reliability scores.

(Step 6) Set different $σ$ shown in (4) at Step 2 and repeat from Step 2 to Step 5. Let $M\_{ni}$ be the number of text descriptions that match with *i*-th candidate code under *n*-th different $σ, n=1,\cdots ,Q$ and let $M\_{1i}$ be the number of text descriptions that match with *i*-th candidate code under the use of $\overbar{p}\_{j\_{l}k}^{(0)}$ in Step 1. Then, we show the difference of classification accuracy compared to the normal classification as

 $d\_{ni}= {M\_{ni}}/{M\_{1i}} , n=1,\cdots ,Q, i=1,\cdots ,L. (5)$$d\_{ni}= {M\_{ni}}/{M\_{0i}}$

# Results

The classifier [1] is applied to *the Family Income and Expenditure Survey* dataset. We randomly extracted 11,000 text descriptions of foodstuff and dining-out data from the dataset and assigned 11 classification codes to the dataset. We used 10,000 instances for training and 1,000 text descriptions for evaluation. Each text description of dataset comprises an item name, specifically a foodstuff name, a food product name, or an item on a restaurant menu in Japanese and a corresponding classification code. We performed code assignments under several conditions (See Table 1).

Table 1. Conditions of code assignment



Table 2. Classification accuracy under each condition





**Figure 1. Robustness of solutions of proposed classifier**

Table 2 shows the classification accuracy of the classifier under each condition. The number of text descriptions that match with the 1st candidate code gradually decreases as the value of $σ$ becomes larger. Meanwhile, the numbers of text descriptions that match with the 2nd and 3rd candidate codes are increasing under that situation. Although the number of total text descriptions that match candidate codes is decreasing as the value of $σ$ becomes larger, it decreases very gradually. Figure 1 shows values of $d\_{ni}$ shown in (5) with respect to values of $σ$. In this figure, the blue line shows the values of $d\_{n1}$, the red line shows the values of $d\_{n2}$, the green line is the values of $d\_{n3}$, and the purple line is the case when we use all of the candidates. From this figure, it can be seen that generally the solutions of the classifier have robustness, in particular, the solutions for the 1st candidate has almost perfect stability for the robustness of the solutions.

# Conclusions

This abstract illustrated an overview of the investigation on the robustness of our previously proposed classifier based on reliability score [1]. The numerical examples show the robustness. We will introduce in-depth analysis of the robustness of our classifier including another numerical example that uses short English text description data in our presentation and subsequent paper.

# References

[1] Y. Toko, S. Iijima, M. Sato-Ilic, Overlapping classification for autocoding system, Journal of Romanian Statistical Review, Vol. 4, (2018), 58-73.

[2] W. Hacking and L. Willenborg, Coding; interpreting short descriptions using a classification, Statistics Methods (2012), Statistics Netherlands, The Hague, Netherlands, Available at: https://www.cbs.nl/en-gb/our-services/methods/statistical-methods/throughput/throughput/coding (accessed August 2018).

[3] H. Gweon, M. Schonlau, L. Kaczmirek, M. Blohm, S. Steiner, Three methods for occupation coding based on statistical learning, Journal of Official Statistics, Vol. 33, No. 1, (2017), 101-122.

[4] Y. Toko, K. Wada, M. Kawano, A supervised multiclass classifier for an autocoding system, Journal of Romanian Statistical Review, Vol. 4, (2017), 29-39.

[5] H. Tsubaki, K. Wada, Y. Toko, An extension of Taguchi's T method and standardized misclassification rate for supervised classification with only binary inputs, presented in poster session in the ANQ Congress, Kathmandu, Nepal (2017).

[6] T. Shimono, K. Wada, Y. Toko, A supervised multiclass classifier using machine learning algorithm for autocoding, Research Memoir of Official Statistics, Vol. 75, (2018), 41-60 (in Japanese).

[7] Y. Toko, K. Wada, S. Iijima, M. Sato-Ilic, Supervised multiclass classifier for autocoding based on partition coefficient, Intelligent Decision Technologies 2018, Smart Innovation, Systems and Technologies, Springer, Switzerland, Vol. 97, (2018), 54-64.

[8] J. C. Bezdek, J. Keller, R. Krisnapuram, N.R. Pal, Fuzzy Models and Algorithms for Pattern Recognition and Image Processing, Kluwer Academic Publishers, New York (1999).

 [9] T. Kudo, K. Yamamoto, Y. Matsumoto, Applying conditional random fields to Japanese morphological analysis, in the 2004 Conference on Empirical Methods in Natural Language Processing on proceedings (2004), 230-237.