Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 1611–1617
Marseille, 11–16 May 2020

Semi-Automatic Construction and Refinement of an Annotated Corpus for a Deep Learning Framework for Emotion Classification

Jiajun Xu[1,2], Kyosuke Masuda[2], Hiromitsu Nishizaki[2], Fumiyo Fukumoto[2], Yoshimi Suzuki[2]
[1]Hangzhou Dianzi University, Hangzhou, Zhejiang, 310018, China
[2]University of Yamanashi, 4-3-11 Takeda, Kofu-shi, Yamanashi 400-8511 Japan
{jiajunxu,kyosuke}@alps-lab.org, {hnishi, fukumoto, ysuzuki}@yamanashi.ac.jp

Abstract

In the case of using a deep learning (machine learning) framework for emotion classification, one significant difficulty faced is the requirement of building a large, emotion corpus in which each sentence is assigned emotion labels. As a result, there is a high cost in terms of time and money associated with the construction of such a corpus. Therefore, this paper proposes a method of creating a semi-automatically constructed emotion corpus. For the purpose of this study sentences were mined from Twitter using some emotional seed words that were selected from a dictionary in which the emotion words were well-defined. Tweets were retrieved by one emotional seed word, and the retrieved sentences were assigned emotion labels based on the emotion category of the seed word. It was evident from the findings that the deep learning-based emotion classification model could not achieve high levels of accuracy in emotion classification because the semi-automatically constructed corpus had many errors when assigning emotion labels. In this paper, therefore, an approach for improving the quality of the emotion labels by automatically correcting the errors of emotion labels is proposed and tested. The experimental results showed that the proposed method worked well, and the classification accuracy rate was improved to 55.1% from 44.9% on the Twitter emotion classification task.

Keywords: emotion classification, construction of emotion corpus, deep learning, dataset refinement, Twitter

1. Introduction

Recently, various cognitive problems, such as image (Deng et al., 2009) and speech recognition (Hinton et al., 2012), have been solved by deep learning-based machine learning frameworks. A deep learning-based cognitive system needs a large number of datasets for model training to prevent the model from over-training (over-fitting). A dataset for model training usually consists of a pair of input data and its annotated teacher label. However, the construction of a dataset is highly costly because in the past humans have usually had to annotate the teacher labels to the dataset. This has resulted in more cost-effective proposals of data augmentation methods (Cubuk et al., 2019; Nishizaki, 2017; Kafle et al., 2017) being put forward. Although it is comparatively easy to perform data augmentation for image and audio datasets, data augmentation for a text corpus for emotion classification is considerably more difficult.

This paper describes an emotion classification of text sentences from Twitter [1], one of the most popular social networking services. To deal with an emotion classification task, a training dataset has to be prepared and constructed for the emotion classifiers. For example, SemEval-2018 Task 1 (Affect in Tweets)(Mohammad et al., 2018) prepared 8,640 English sentences with manually annotated labels. Moreover, about 30,000 emotion labels were manually attached to text dialogs in the SemEval-2019 Task 3 (EmoContext) (Chatterjee et al., 2019). Alongside the corpora of SemEval, there are several other English text corpora for emotion classification; however, the number of emotion corpora for other languages is limited. For example, one such corpus was that of Saputri et al. (Saputri et al., 2018), who built an Indonesian Twitter dataset for emotion classification. They collected 7,500 tweets and manually annotated emotion labels to these tweets.

On the other hand, Bostan and Klinger (Bostan and Klinger, 2018) investigated and analyzed annotated corpora for emotion classification. They referred to many emotion corpora in their paper, and they also reported that the size of even the largest corpora was fewer than 40,000 labels. With this fact in mind, to build an annotated corpus for emotion classification can be considered an arduous task. However, a defining factor of this study is that it describes a semi-automatically constructed annotated corpus for emotion classification from Japanese tweets; the size of the corpus was about 80,000 labels, which must be considered as one of the largest corpora for emotion classification to date. It is also of note that this approach to building an emotion corpus can easily be applied to other languages.

The construction steps of our corpus are as follows:

1. Emotion words were selected as seed keywords from the emotion categorization dictionary (Nakamura, 1993). Each seed emotional keyword belongs to an emotion class. This was the only part of the process performed by a human, all the other steps were automated.

2. Tweets were automatically collected by searching a seed emotional keyword. The tweets were annotated with the same emotion label to the seed emotional keyword automatically. This sometimes led to the tweet being assigned the wrong emotion label. This was used as an initial dataset for the emotion classifier training.

3. The emotion classifier was trained using the initial training dataset.

4. Emotion labels were automatically updated by the emotion classifier.

[1]`https://twitter.com/`

5. Steps 3. and 4. were repeated.

By repeating the process described above, the initial dataset would eventually become more refined. It is hard to apply a data augmentation approach to an emotion corpus in which the data size is small. Therefore, we proposed the approach of semi-automatic construction of emotion corpus and refinement of emotion labels.

There have been some studies released for similar approaches to dataset refinement or distillation. For example, Cosentino and Zue (Cosentino and Zhu, 2019) proposed a re-labeling approach using a Generative Adversarial Network (GAN), and the approach was evaluated on the MNIST task (Lecun et al., 1998). First, a numeric image was inferred to the neural network-based classification model, and the classification result was judged and either accepted or rejected based on a confidence measure. If the input image was rejected, the image was regenerated by the GAN. This refinement approach is quite similar to the refinement approach detailed above, even though the target media is different, meaning that the approach applied here may be simpler. On the other hand, Köhler et al. (Kohler et al., 2019) proposed a detection and filtering approach for noisy labels on the CIFER-100 image dataset. This approach estimated uncertainty measures for an input image, and uncertain images that were detected were removed. The approach followed in this paper does not need any calculation of uncertainty measures for tweet sentences.

The contribution of this paper is first to show that the refinement approach of emotion corpus works effectively. In addition to this, it also describes the best model selection approach using almost the same approach as the refinement process of the training dataset.

The refinement approach used for the dataset in this study was automatically built from Twitter and was evaluated with an emotion classification task wherein human subjects annotated emotion labels to the sentences in the test set. The experimental results showed that the classification accuracy rate of the model trained from the refined training dataset by our refinement approach largely surpassed that of the baseline model trained from the initial training dataset, including noisy labels. Overall, a 10.2% improvement of the accuracy rate was achieved. On the other hand, unfortunately, the best model selection approach did not work as well, and it is evident that it has much room for improvement.

This paper is organized as follows: Section 2 explains the (initial) dataset construction and neural network architectures used in the emotion classification task. Section 3 describes the refinement process of the training set for the emotion classifier. Section 4 shows the experimental setups for the emotion classification task and its results, and conclusions are drawn in Section 5.

2. Emotion Classification using Deep Learning

In this paper, we deal with eight sorts of emotion categories based on the Plutchik's emotion wheel (Plutchik, 1980); joy, sadness, trust, disgust, fear, anger, anticipation, and surprise.

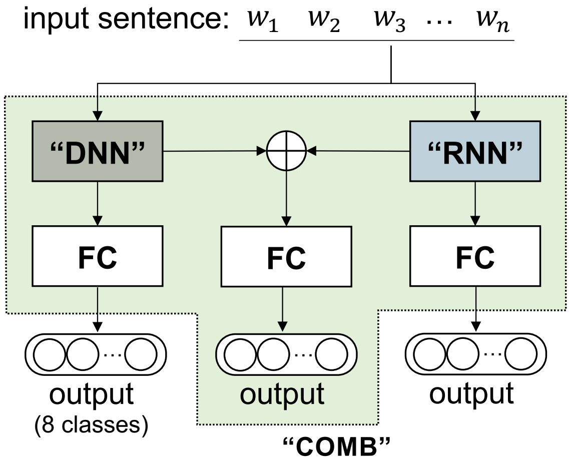

Figure 1: Structure overview of the emotion classification model.

2.1. Initial Training Set for the Emotion Classifier

First, a semi-automatically constructed initial training dataset was used for training the emotion classifier. The training data for the emotion classifier consists of the sets of a pair (S_1, L_1), where S_1 and L_1 refer to a sentence and its emotion label, respectively. The steps of building the initial training data are explained as follows.

1. One hundred and twenty-four "seed emotional keywords" were selected for each emotion category from the "Emotion Representation Dictionary" (Nakamura, 1993). We assumed that the dictionary lists representative emotional keywords that have strong ties to each emotion category. Table 1 shows a part of the seed emotional keywords and their corresponding emotion category. Therefore, each seed emotional keyword can be connected to one emotion category.

2. Sentences were collected from Twitter using the 124 seed keywords. Each sentence was labeled depending on the emotion category of the seed keyword.

3. Each sentence was segmented into words using the Japanese morphological analyzer MeCab (Kudo et al., 2004).

4. Ten thousand tweets (sentences) were randomly selected for each of the eight emotion categories. These were then used to construct the initial training data that contained (S_1, L_1),..., (S_N, L_N), where N = 80,000.

As described above and in Table 1, a tweet (sentence) was automatically annotated with one emotion label based on the hypothesis that a seed emotional keyword is strongly connected to one emotion category. Therefore, the initial training dataset may contain many wrongly labelled sentences. This is because the emotion category of a seed emotional keyword and the true emotion category of the searched sentence by the keyword do not always coincide.

2.2. Emotion Classification Model

In this study, we use neural network-based models for emotion classification. Figure 1 shows an outline structure of

emotion category	emotional keywords
joy	happy, interesting, thankful celebration, enjoy, proud
trust	respect, calmness, reliable love, comfortable, welcome
fear	horrible, terrible, lonely dark, suspect, scaring
surprise	unexpectedly, unbelievable, great dissappar, no way, impressed
sadness	sad, disappointment, lonely unfortunate, remorseful, poor
disgust	rammy, croosh, awkward depressing, gloomy, imperfect
anger	offensive, exasperated, sore loud, sleepy, angly
anticipation	hopeful, wary, tension excited, fluttery, atwitter

Table 1: Examples of seed keywords for searching emotional tweets

the neural network-based emotion classification model used in this study. There are some excellent models such as Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019), which achieved superior results on various natural language processing tasks. However, we did not try to use state-of-the-art neural network architectures because the purpose of this paper is to make a training corpus (sets of a sentence and their emotion labels) for a robust emotion classifier semi-automatically. Therefore, our proposed method for making a training dataset does not depend on any specific neural network architectures.

As shown in Figure 1, the neural network-based model is composed of the three following main model architectures: one is a model consisting of two fully connected (FC) layers only (denoted as "DNN" model in this paper), the second one is composed of a recurrent layer 2 and an FC layer (denoted as "RNN" model), and the final one is the combination of both the outputs of the DNN and RNN (denoted as "COMB"). The output of COMB is used as an emotion classification result.

The details of the DNN and RNN models are shown in Figures 2 and 3. The neural network used adopts a backpropagation considering multiple losses; in other words, the three losses were calculated based on the outputs of the DNN, RNN, and COMB models, and then these loss values were averaged. This average was then used for the backpropagation process. In previous trials, it was found that each model was good at the specific emotions[2]. Therefore, this study adopted the multiple losses calculation so as to take advantage of the characteristics of each model.

2.2.1. DNN Model

Figure 2 shows the neural network architecture of the DNN model, which is a part of the whole network. The input is a sentence W, and W consists of n words. Each word (represented as a one-hot vector) is input into the embedding layer, and it converts a 128-dimensional vector. Each word-embedding vector proceeds to the FC layer. All the outputs (256 dim.) for input words are averaged. This averaged vector is regarded as a sentence-embedding vector in this paper.

2.2.2. RNN Model

Figure 3 shows the neural network architecture of the RNN model. The RNN model's architecture is almost the same as that of the DNN model. The difference from the RNN model is that an LSTM layer is used instead of the FC layer between the word-embedding layer and the average layer. The LSTM layer can consider the context information of the word sequence of the input sentence. As with the RNN model, each output vector of the LSTM layer is averaged against input words and is used as a sentence-embedding vector.

2.2.3. COMB Model

Figure 1 is the whole model architecture of the combination of the DNN and RNN models. The output vectors from DNN and RNN are concatenated and then proceed to the FC layer. In addition, softmax cross-entropy is used as a loss function. As with the COMB model, the DNN and RNN models also calculate cross-entropy losses and these three losses are used for the backpropagation.

3. Refinement of the Training Dataset and the Training Emotion Classifier

In this study, a refinement approach is proposed as a way of developing the training dataset for the emotion classifier. Figure 4 shows the process of the proposed method for the training dataset refinement, in which the noisy pairs of a sentence and its emotion label are automatically removed from the training dataset. In addition, Figure 5 shows the process of training and selecting of the best emotion classifier model and the re-labeling process (as seen in in Figure 4).

3.1. Refinement of the Training Set

The refinement process for a training set comprises the following three steps: the first step is the training of an emotion classification model, the second is the re-labeling of the emotion label of each sentence in the training dataset, and the final step is the removal of noisy pairs.

In the first step, we train the emotion classification models by L repetitions with the re-labeling of the emotion label for each sentence in the training dataset. The architecture of the emotion classification model as shown in Figure 1 and Table 2 also shows the model training condition. The l^{th} model is trained with the revised training dataset, which is made by the previous training loop. Note that we use the initial training dataset when $1 = 1$. The number of epochs for all the models is set to 20. We evaluate the model in each epoch against the validation set, in a sentence and its emotion label are not included in the training dataset. The best epoch model for the validation set is selected and used for the re-labeling process.

The second step is the re-labeling process. The selected model (l^{th} trained model) classifies the emotion category

[2]For example, the DNN model was good at "joy" emotion.

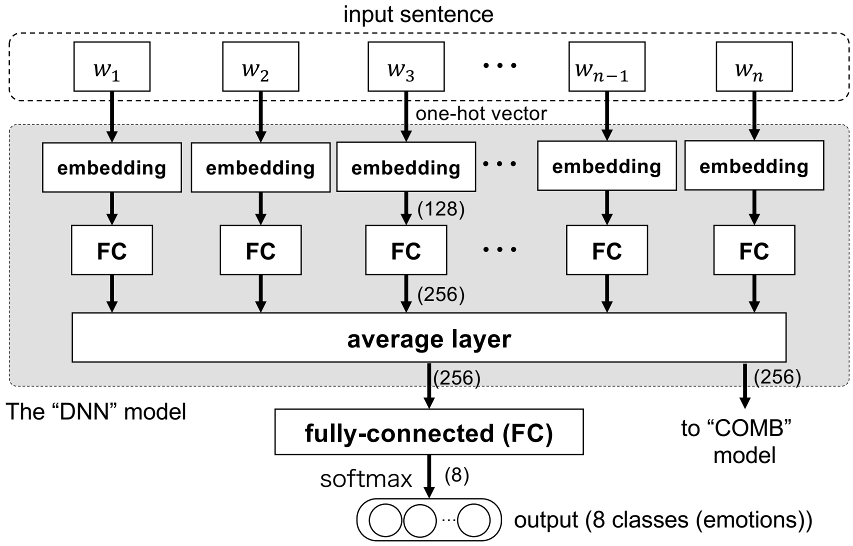

Figure 2: Model architecture of the DNN model.

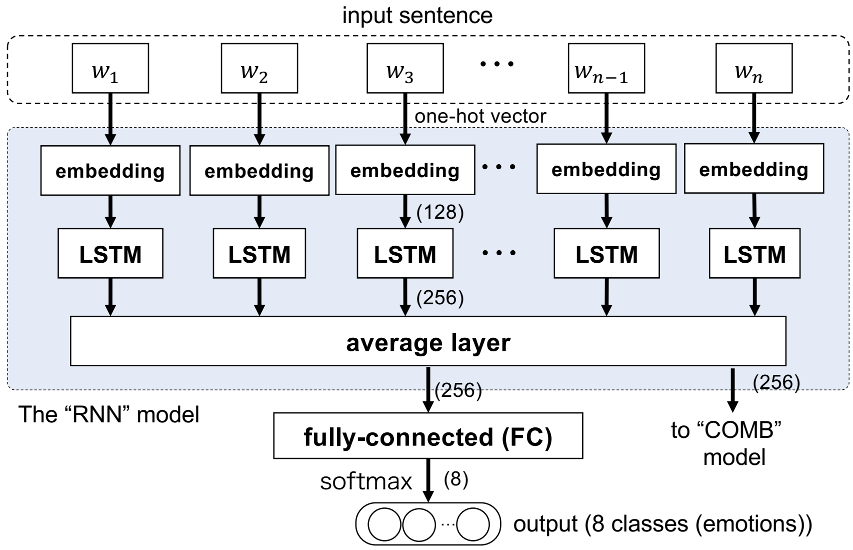

Figure 3: Model architecture of the RNN model.

against each sentence in the $l-1^{th}$ revised training set. When the emotion label from the model is different from the original label for a sentence, the emotion label is changed to the new one. After all the sentences in the training set are re-labeled, the new l^{th} revised training set is saved.

In this research, the first and second steps were repeated L times, where $L = 100$. After finishing L repetitions, the noisy sentence–emotion label pairs were removed from the initial training dataset. The $L+1$ training datasets were already stored after the training and re-labeling L loops. In this study, any sentence with an emotion label that was replaced at least once during the re-labeling process was regarded as a "noisy" sentence. If an emotion label fluctuates depending on the model for a sentence, it is hard to classify the emotion category clearly for said sentence. Alternatively, if the original emotion label of the sentence is entirely wrong, the noisy sentence and its emotion label are removed from the initial training dataset. Finally, a refined training dataset was obtained, which was used for the best emotion classification model training.

Dim. of word embedding	128
Mini-batch size	128
Num. of epochs	20
Activation at FC layers	ReLU(Glorot et al., 2011)
Dropout	0.2 at all the hidden layers
Loss func.	Softmax cross entropy
Optimizer	Adam(Kingma and Ba, 2015)
Init. learning rate	0.001

Table 2: Training conditions of the models

3.2. Training of the Best Emotion Classifier

The training procedure of the emotion classifier is the same as that of the refinement of the training dataset described in the previous section. Figure 5 shows the training process, which is the same as that in Figure 4 and includes the re-

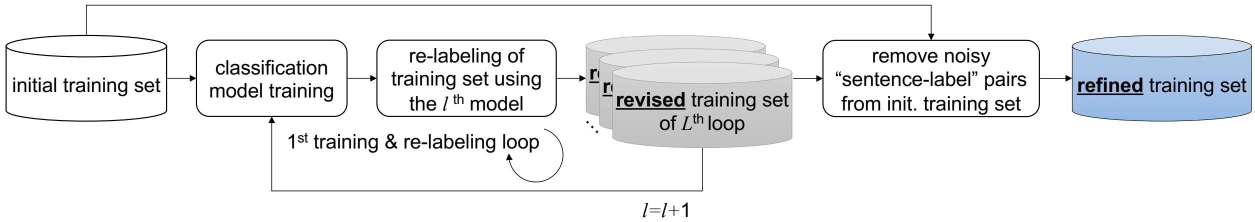

Figure 4: Refinement procedure of the sentence–emotion pairs from the initial training dataset.

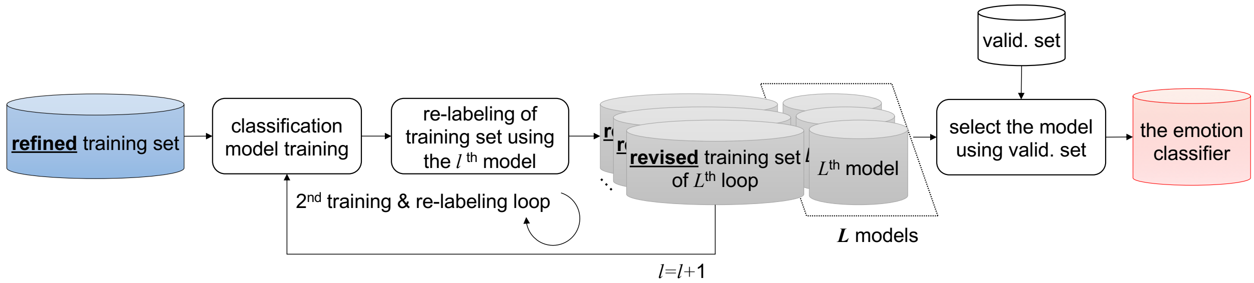

Figure 5: Procedure of the training of the best emotion classifier.

labeling procedure.

The emotion classification models were trained L times again using the revised training dataset. The same re-labeling process was also applied during model training. After finishing the training and re-labeling loops, L classification models were gained. From the L models, the emotion classifier that obtained the best performance on the validation set was chosen. The best classifier is evaluated on the test set for the evaluation of our proposed method.

4. Experiment

4.1. Experimental Setup

In this paper, the proposed method was evaluated in sentences from Twitter. As mentioned in Section 2.1, 80,000 sentences were collected from Twitter. They were used as the initial training dataset for training the model. Also, an additional 480 sentences were included. Six people manually annotated the emotion labels to these tweet sentences. If more than two people annotated the same emotion label to the same sentence, the emotion label was adopted. Finally, 221 pairs of sentences were prepared and their emotion labels for validation were added and 207 pairs for the testing were similarly prepared, respectively.

Using the eight emotion categories based on the Plutchik's emotion wheel (Plutchik, 1980), the emotion classifiers were evaluated on the classification accuracy rate.

4.2. Experimental Results and Discussion

Table 3 shows a summary of the emotion classification accuracy rates and Figure 6 shows the classification accuracy rates on the test set at each training and re-labeling loop. The "Baseline" is the result of the initial model trained with the initial training dataset without the re-labeling loop being included. The "Refined train. set" comes from the model trained with the refined training dataset without including the re-labeling loop. In Figure 5, "The selected model" is the result of the selected model from the L models based on the validation set. "The best performance model" is the oracle model among the L models, which obtained the best performance on the test set.

The "Baseline" had the worst accuracy rate because the initial training dataset included many wrong sentence–emotion label pairs. As described in Section 2.1, the initial training dataset was semi-automatically built using the 124 seed emotional keywords. One-hundred sentence–emotion label pairs were randomly selected from the training dataset and evaluated by a human. As a result, 17% of the sentences were found to be mistakenly labeled. However, the refinement process for the initial training dataset successfully removed the noisy pairs from the initial training dataset. The error rate of the emotion label annotations on the refined training dataset was 13%, which amounted to a 4% reduction. Therefore, the classification accuracy rate of the "Refined train. set" was drastically improved to 55.1% from the baseline model. From the test set accuracy on the second training loop in Figure 6, it can be seen that almost all models based on the refined training dataset were improved. These results are put forward as evidence that the proposed refinement approach in this study for the training dataset efficiently worked on the training of a better classifier.

On the other hand, the second training and re-labeling loop did not work as well because "The selected model" was slightly worse (54.6%) than the "Refined train. set." After the refinement of the initial training dataset, wrong emotion labels were removed. Therefore, the re-labeling loop did not contribute to the improvement of the classifier. However, "the best performance model" obtained a 58.5% classification accuracy rate. This indicates that the second re-labeling loop can improve the classifier. That said better model selection criteria from those of the L models will be put forward for future development.

5. Conclusions

This paper has described the semi-automatic construction approach of an annotated corpus from Twitter for emotion

Baseline (initial train. dataset)	44.9
Refined train. set	**55.1**
The selected model	54.6
The best performance model	58.5

Table 3: Emotion classification accuracy rates [%]

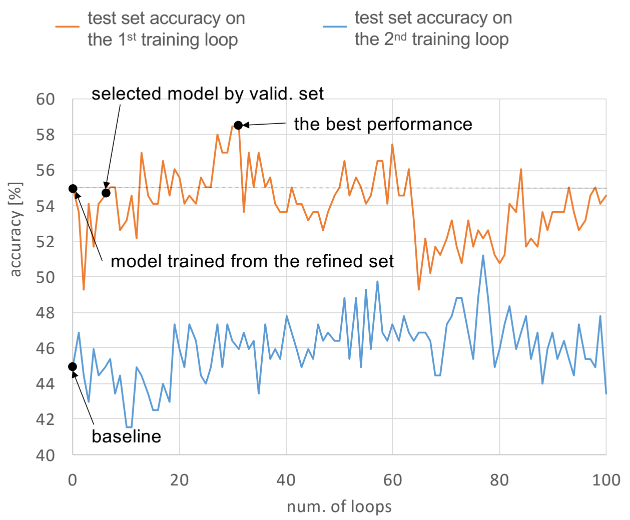

Figure 6: Classification accuracy rates on the test set at each training and re-labeling loop.

classification. The approach outlined seldom needs human annotation of emotion labels, and this paper showed that our approach could semi-automatically build the corpus with annotated emotion labels with greater efficiency than previous models. The approach was simple. First, tweet sentences were collected by searching seed emotional keywords on Twitter. Human work amounts only to the preparation of the seed keywords. Second, the initial model for emotion classification was trained with the initial training set with noisy emotion labels. The model was used for re-labeling emotion labels on the training set. The model training and the re-labeling processes were repeated many times. Finally, the refined training set for the emotion classifier was achieved. The best model selection method, which is the same procedure as that of the dataset refinement, was also proposed.

The experimental results showed that the proposed approach worked well, and the classification accuracy rate was improved to 55.1% from 44.9%, from the baseline model without any refinement processes, on the Twitter emotion classification task. On the other hand, the best model selection method did not work as well because it entirely depends on the validation set. However, a 58.5% classification accuracy rate was obtained if the best model was selected. Therefore, there is a requirement for further develop the selection method of the best model from multiple models.

In future works, we are going to use the start-of-the-art neural network models on this task. In addition, our methods will be applied to other languages such as Chinese and English.

6. Acknowledgements

This work was supported by JSPS KAKENHI Grant-in-Aid for Scientific Research (B) Grant Number 17H01977. Besides, a part of this work was also supporoted by Hoso Bunka Foundation.

7. Bibliographical References

Bostan, L.-A.-M. and Klinger, R. (2018). An analysis of annotated corpora for emotion classification in text. In *Proc. of the 27th International Conference on Computational Linguistics*, pages 2104–2119.

Chatterjee, A., Narahari, K. N., Joshi, M., and Agrawal, P. (2019). SemEval-2019 task 3: EmoContext contextual emotion detection in text. In *Proc. of the 13th International Workshop on Semantic Evaluation*, pages 39–48.

Cosentino, J. and Zhu, J. (2019). Generative well-intentioned networks. In *Proc. of the33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, page 12.

Cubuk, E. D., Zoph, B., Mané, D., Vasudevan, V., and Le, Q. V. (2019). Autoaugment: Learning augmentation strategies from data. In *Proc. of CVPR 2019*, pages 113–123.

Deng, J., Dong, W., Socher, R., Li, L., Kai Li, and Li Fei-Fei. (2009). Imagenet: A large-scale hierarchical image database. In *Proc. of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL-HLT 2019*, pages 4171—-4186.

Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep Sparse Rectifier Neural Networks. In *Proc. of the 14th International Conference on Articial Intelligence and Statistics (AISTATS) 2011*, pages 315–323.

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., and Kingsbury, B. (2012). Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups. *IEEE Signal Processing Magazine*, 29(6):82–97, Nov.

Kafle, K., Yousefhussien, M., and Kanan, C. (2017). Data augmentation for visual question answering. In *Proc. of the 10th International Conference on Natural Language Generation*, pages 198–202.

Kingma, D. P. and Ba, J. L. (2015). Adam: A method for stochastic optimization. In *Proc. of the 3rd International Conference on Learning Representations (ICLR 2015)*, page 15 pages.

Kohler, J. M., Autenrieth, M., and Beluch, W. H. (2019). Uncertainty based detection and relabeling of noisy image labels. In *Proc. of CVPR Wrokshop 2019*, pages 33–37.

Kudo, T., Yamamoto, K., and Matsumoto, Y. (2004). Applying conditional random fields to japanese morphological analysis. In *Proc. of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pages 230–237.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11):2278–2324.

Mohammad, S., Bravo-Marquez, F., Salameh, M., and Kiritchenko, S. (2018). SemEval-2018 task 1: Affect in tweets. In *Proc. of The 12th International Workshop on Semantic Evaluation*, pages 1–17.

Nakamura, A. (1993). *Emotion Representaion Dictionary*. Tokyodo Publishing.

Nishizaki, H. (2017). Data augmentation and feature extraction using variational autoencoder for acoustic modeling. In *Proc. of the 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 1222–1227.

Plutchik, R. (1980). A general psychoevolutionary theory of emotion. *Theories of emotion*, 1(4):3–31.

Saputri, M. S., Mahendra, R., and Adriani, M. (2018). Emotion classification on indonesian twitter dataset. In *Proc. of the 2018 International Conference on Asian Language Processing (IALP)*, pages 90–95.