

ASCII Art Classification based on Deep Neural Networks Using Image Feature of Characters

Kazuyuki Matsumoto^{1*}, Akira Fujisawa², Minoru Yoshida¹, Kenji Kita¹

¹ Tokushima University, Tokushima, Japan.

² Aomori University, Aomori, Japan.

* *Corresponding author. Tel.: +81886567654; email: matumoto@is.tokushima-u.ac.jp

Manuscript submitted August 31, 2018; accepted October 10, 2018.

doi: 10.17706/jsw.13.10.559-572

Abstract: In recent years, a lot of non-verbal expressions have been used on social media. Ascii art (AA) is an expression using characters with visual technique. In this paper, we set up an experiment to classify AA pictures by using character features and image features. We try to clarify which feature is more effective for a method to classify AA pictures. We proposed five methods: 1) a method based on character frequency, 2) a method based on character importance value and 3) a method based on image features, 4) a method based on image features using pre-trained neural networks and 5) a method based on image features of characters. We trained neural networks by using these five features. In the experimental result, the best classification accuracy was obtained in the feed forward neural networks that used image features of characters.

Key words: ASCII art, deep neural networks, classification, image feature, character feature

1. Introduction

ASCII art is an expression that is often used on electronic bulletin board or on other Internet communication as a non-verbal expression. Because ASCII arts are multiline expressions, therefore, character-based analysis of ASCII arts is more difficult compared to emoticons. However, meanings and contents that can be expressed by an ASCII art are very versatile, and ASCII art is certainly important expression that should not be ignored for web data analysis. ASCII art expresses a picture visually by using characters instead of dots or lines, so each character used in ASCII art does not have any sense except for words spoken by a character in the ASCII art picture or captions for the picture. Therefore, it is more suitable to treat ASCII art as an image than as a set of characters.

In this paper, to validate whether image features are effective for category classification of ASCII art, we create ASCII art category classifiers by training character features (character appearance frequency, character importance) and image features obtained by imaging the ASCII art. By evaluating these classifiers with experiments, we would like to discuss effective features for ASCII art classification.

The examples of ASCII art are shown in Fig. 1. Generally, ASCII arts are able to classified into two types; i) ASCII arts which are created from source original image, ii) ASCII art which are created originally. In Fig.1, upper left example ASCII art is created in the motif of existing mascot character in the right side. Lower example ASCII arts are created originally in an anonymous bulletin board website “2-channel.”

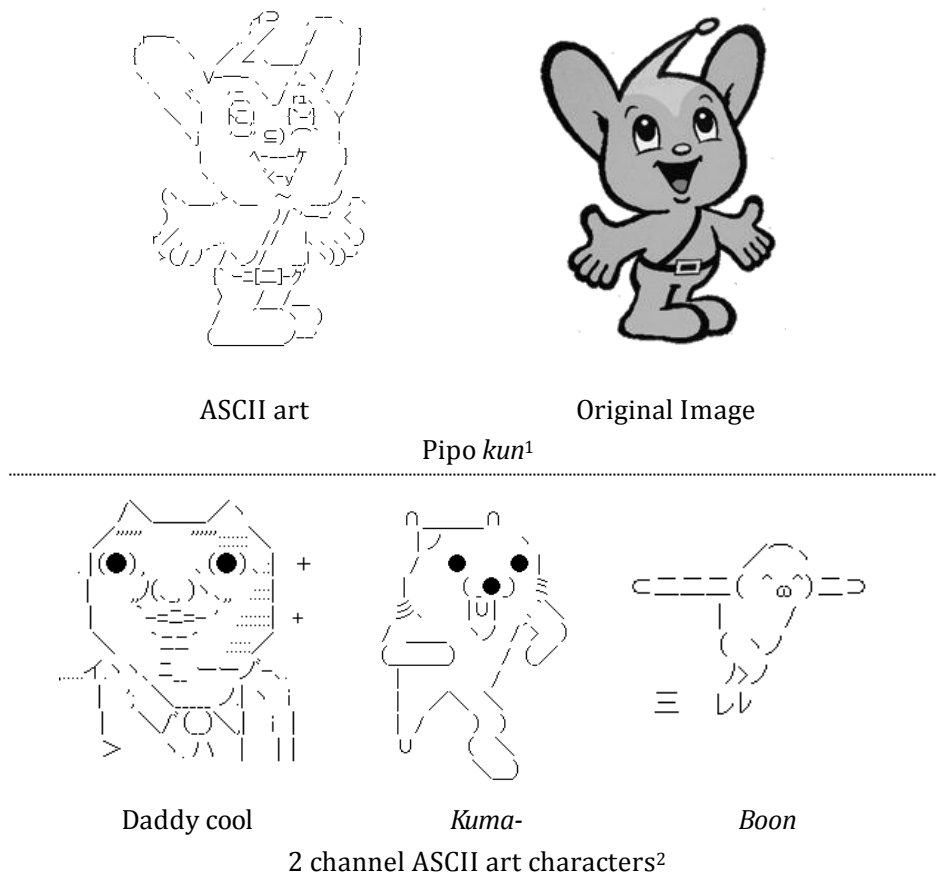


Fig. 1. Examples of ASCII art.

2. Related Research

2.1. Extraction / Detection of ASCII Art

Tanioka *et al.* [1] proposed a method to classify texts into the ones including ASCII art and others without ASCII art by using a support vector machine based on byte patterns and the morphological analysis result. Their method achieved 90% of classification accuracy.

Hayashi *et al.* [2] proposed a method to extract ASCII art independent from language types. They focused on the frequency when the same character appeared twice in series. Then, they used LZ77 compression ratio and RLE compression ratio as feature. As the result, their method achieved over 90% of extraction F-value. Suzuki *et al.* [3] focused on compression ratio as same with Hayashi *et al.* They used C4.5 as machine learning algorithm and their extraction method was also independent from language types.

Both studies focused on binary classification task to judge if they are ASCII arts or not, and their methods did not focus on understanding the meanings of the ASCII arts. In this point, their studies are different from our study.

2.2. Classification of ASCII Art

Yamada *et al.* [4] proposed the classification algorithm of emoticon based on character N-gram feature. Their method could obtain high accuracy of emotion classification.

Matsumoto *et al.* [5] proposed a method to classify emoticon into emotion category by using character

¹ http://www.keishicho.metro.tokyo.jp/about_mpd/shokai/pipo/profile.html

² <http://kenji1234.blog75.fc2.com/>

embedding feature. Their method could better performance than the baseline method using character N-gram feature based on machine learning algorithms such as SVM, logistic regression, etc.

Fujisawa *et al.* [6] classified emoticon into emotion category by using image feature such as histograms oriented gradients, local binary patterns, etc. Their method used k-nearest neighbor classifier as a machine learning algorithm. The distance/similarity calculation is Levenshtein Distance (LD) and cosine similarity. Jirela *et al.* [7] proposed the method to classify emoticon image into eight emotion categories by using deep convolutional neural networks. In their study, to make up for the lack of the number of emoticon data, they used image data augmentation by using other font types and image conversion such as rotate, flip, and noise addition.

Fujisawa *et al.* [8] proposed a method to evaluate ASCII art similarity by using image features. In their study, ASCII arts were converted into images. By using image features, they succeeded in confirming similarity between the image features of the ASCII arts that were made based on emoticons, without any influence from the differences of characters or sizes.

Fujisawa *et al.* [9] compared the similarity degrees of both image features extracted from emoticons and from large-sized ASCII arts that were made based on the emoticons. Their paper described that the shape feature HOG (Histograms Oriented Gradients) is effective as an image feature.

In this paper, we construct ASCII art category classifiers using character features and image features. We compare the methods to classify ASCII art categories by evaluating the constructed classifiers.

2.3. Generation of ASCII Art

Xu *et al.* [10] proposed a novel structure-based ASCII art generation method. Generally, existing tone-based ASCII art generation method requires high text resolution for display. The method by Xu focused on the structure of ASCII art, it can consider the text resolution by using suitable characters to ASCII art image size.

Takeuchi *et al.* [11] have focused on a method to efficiently generate ASCII art. Because large scale ASCII arts are generally very complex, it is very difficult to automatically generate ASCII art with high quality. They realized a high-speed generation of ASCII art from the original images by reduction of the computing time using local exhaustive search. There is a project which used deep learning method for generation ASCII art from line art image [12].

Those approaches can generate ASCII arts, however, their studies do not focus on ASCII art category classification. It is considered that ASCII arts are various type and it is more difficult to define category than the picture or illustration image. The dataset of ASCII art is not easy to construct because each ASCII art of author is not clear, and the authorship is not clear. Thus, if we use ASCII art generation algorithm, we maybe can overcome that problems. However, this paper does not describe about ASCII art generation, we think that those are very important technique.

3. Feature Extraction from ASCII Art

3.1. ASCII Art Classification by Using Image Features

We extract image features from a full picture of ASCII art, then train the features using machine learning to make an ASCII art classifier. To convert an ASCII art into an image, we use AAtoImage[13]. The ASCII art converted into image is normalized as an image with NxN pixels size and converted into an edge image by binarization. In this paper, we set N=128.

Following the above procedure, we prepared edge images for each category. Then, the edge images were trained by convolutional neural networks to make an ASCII art classifier. To avoid effects from training bias due to the number of the images, we adjusted the number of the training data in advance not to cause bias

between the categories. We also proposed a method based on Histograms Oriented Gradients(HOG)[14] features extracted by character unit.

The flow of making an ASCII art classification model based on HOG features of character images is as follows:

All of characters included in the ASCII arts of the training data are converted into images.

The character images are converted into the edge images to extract HOG features. The hyper parameters of HOG features are shown in Table 1.

Table 1. Parameters of HOG Features

Parameters	Value
Orientations	9
Pixels per cell	(8,8)
Cells per block	(3,3)

HOG features are clustered by a repeated bisection method [15]. The number of clusters k is set as 1000.

Based on the clustering results, the vector indicating the class frequency of the character image is obtained for each ASCII art.

Next, we calculate the class importance values for each character image class. The value is calculated based on the idea of TF*IDF [16]. The calculation formula is shown the equation (1). The calculated values are set as element of vector. $cf_i^{(j)}$ indicates the frequency of character image class i in the ASCII art j . N indicates the total number of the ASCII arts in the training data. dcf_i indicates the number of ASCII arts which character image class i appeared in. if dcf_i indicates zero value, we set $iv_i^{(c)}$ as 0.

$$iv_i^{(c)} = cf_i^{(c)} \times \log\left(\frac{N}{dcf_i} + 1.0\right) \quad (1)$$

Finally, the vectors are trained with multi-layer perceptron (MLP) to make an ASCII art classifier.

3.2. ASCII Art Classification by Using Character Features

We propose two methods;

A method that trains the features of the character frequency vectors by MLP

A method that trains the matrix of the character importance values based on TF*IDF by CNN

The character importance value (CF*IAF) is calculated by (2), (3), (4). By calculating the character importance value on character unit, the effects like pixel values in picture images are expected.

$$CF * IAF = CF \times IAF \quad (2)$$

$$CF = freq_i^{(c)} \quad (3)$$

$$IAF = \log\left(\frac{N}{AF} + 1.0\right) \quad (4)$$

CF indicates the frequency of character c in AA_i . N indicates the total number of ASCII arts to calculate importance values. AF indicates how many ASCII arts are included in character c . Fig.2 simply shows the process of extracting the character importance matrix from an ASCII art and training it by CNN.

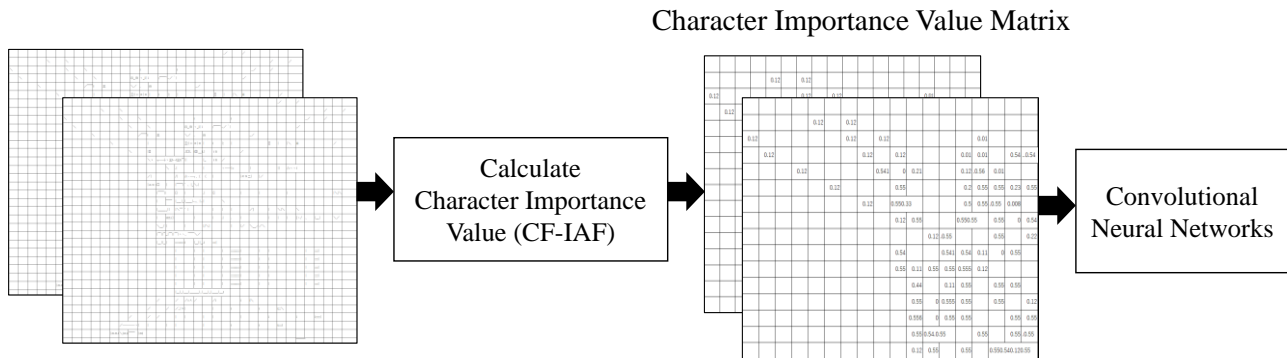


Fig. 2. Extraction of the matrix of character importance from ASCII art.

We set the maximum number of the characters as 100 in a line and the maximum number of the lines as 100 in an ASCII art because the number of the characters and the lines are different depending on the ASCII arts. If the size of ASCII art is smaller than these maximum sizes, we conduct a padding process by using space characters. If the size of ASCII art is larger than these maximum sizes, we extract the features by using the 100 characters from the head of each line and the lines from the first to 100th.

The ASCII arts that have motifs of the characters in the same categories are often created by the same authors, and they usually change only the different parts to create other characters based on the original characters. Therefore, the categories of the ASCII arts can be judged by just using the character frequency feature vectors.

However, even though the characters are in the same categories, they are sometimes the deformed characters or the characters that are not so often created due to their popularity or complexity. Therefore, ASCII art category classification by using character features would be difficult depending on the kinds of ASCII arts.

4. Experiment

4.1. Experimental Setup

To validate how much the image feature based AA classifiers (3 types) and the character feature based AA classifier (2 types) are suitable to ASCII art category classification, we conduct an evaluation experiment on the data created with ASCII arts.

The structure and parameter of each layer are shown in Fig. 3, 4, 5, 6, 7. In training, we set the epoch as 50, and the mini batch size as 256. The filter size in the convolutional layer is 3x3, and the numbers of filters are 20 and 32. We use ReLU as activation function and Adam as optimizer, and set learning rate as 0.0001.

In the training of VggNet, we used pre-trained networks and fine-tuned the VggNet. As input data of VggNet, 3-channel image data is required, however, ASCII arts are converted into grayscale images, so, we converted grayscale image into RGB 3-channel image for training VggNet [17].

The dropout rate in the dropout layer prior to the output layer is set as 0.5, and the Softmax is used as activation function in the output layer. And, we used categorical cross entropy error as the loss function.

We used Keras [18] as the deep learning frontend library, and used TensorFlow [19] as a backend framework. Then, we used early stopping method. The early stopping is a method that stop the training when the loss value is not improved. The maximum iteration number was set as 50.

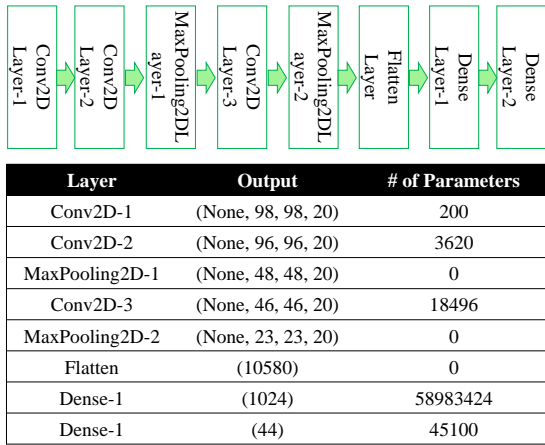


Fig. 3. CNN architecture and parameters (CF-IAF feature).

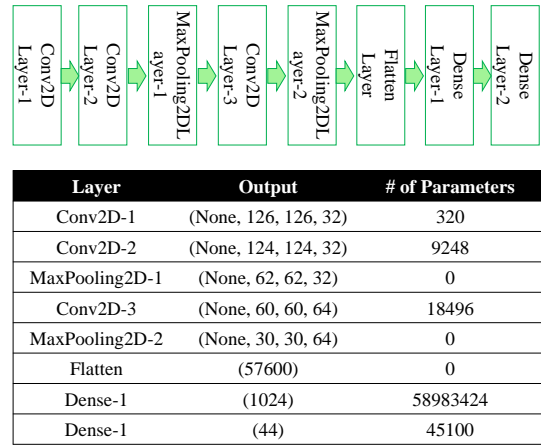


Fig. 4. CNN architecture and parameters (Image feature).

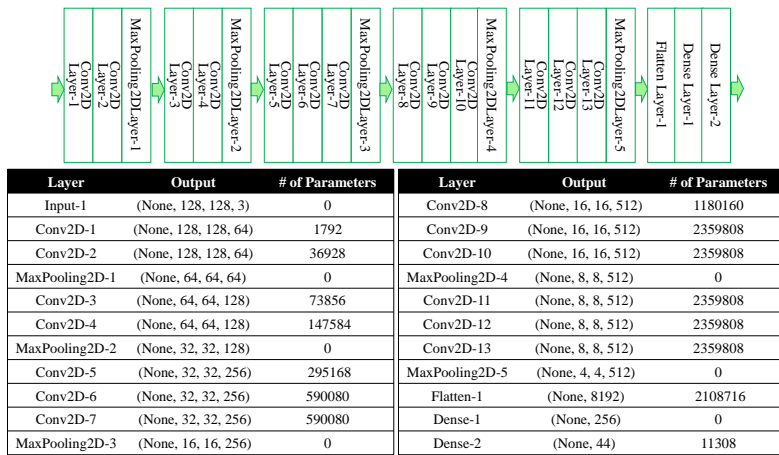


Fig. 5. VggNet architecture and parameters (Image feature).

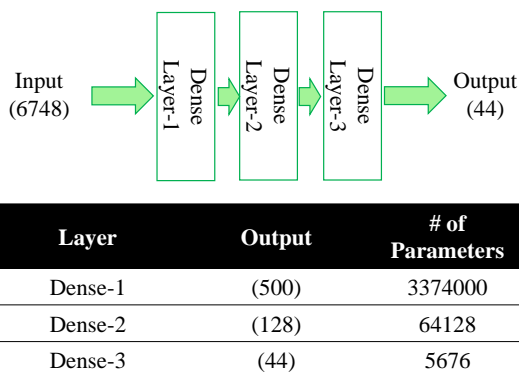


Fig. 6. MLP architecture and parameters (Char. freq. feature).

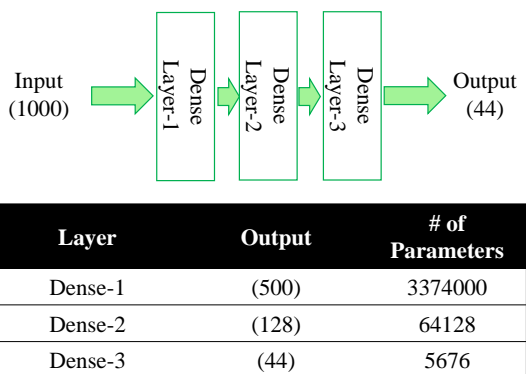


Fig. 7. MLP architecture and parameters (Char. Image feature).

4.2. Data

Our research target on the ASCII arts that are included in the zip file downloaded from the web site: "AAMZ Viewer."³ This data includes a lot of ASCII arts created in the motif of the characters appeared in comics, anime and games. We decided the titles of the comic, anime and game works as ASCII arts categories. We excluded the following ASCII arts from the experimental target: 1) the ASCII arts originally

³ <http://aa.yaruyomi.com/>

created on “anonymity bulletin board: 2-channel” and 2) the ASCII arts that were classified into “other works.”

If these ASCII arts are included for category classification, the ASCII arts with the motif of various works and characters would be included in the same category, they might become noise data for training a classifier. On the other hand, there are serialized works. For example, the all works of Tetsuo Hara have the characters drawn with similar design and style peculiar to Tetsuo Hara. We treat such works as one category (e.g. “Works of Tetsuo Hara”).

We target the 44 categories including over 900 kinds of ASCII arts as experimental data. We divide the training data and the validation data at a rate of 8 to 1.

Both data have the same number of images in each category. The rest of the ASCII arts are used as the evaluation data. Table 2 shows the details of the data.

Table 2. The Number of the Data

Training Data	Validation Data	Test Data
35200	4400	20309

By using the data prepared as above, we create two types of classifiers based on character features and the two types of classifiers based on the image feature.

4.3. Evaluation Method

We look at the probability values for each category outputted by neural network and check if the correct categories are included in the top k=3, 5, 10 of the outputted probabilities. If the correct categories are included, we regarded as successful classification. Then we calculate the classification successful rate. Additionally, we use macro-averaged Precision, macro-averaged Recall, and macro-averaged F1-score for evaluation. Equations (5), (6), (7) indicate each evaluation score. In Eq. (5) and (6), tp_c indicates a true positive, an outcome in which the system correctly estimated category c. In Eq. (5), fp_c indicates a false positive, where the system incorrectly estimated another category as category c. In Eq. (6), fn_c indicates a false negative, an outcome in which the system incorrectly estimated category c.

On the other hand, we regarded as successful classification when the correct categories are outputted in the top 1, and calculated the completely matched rate (accuracy). Besides, we calculate the averaged rank of the cases where classification was successfully made with k=10 to compare the performances of the classifiers.

$$\text{Precision}_c(\%) = \frac{tp_c}{tp_c + fp_c} \times 100 \quad (5)$$

$$\text{Recall}_c(\%) = \frac{tp_c}{tp_c + fn_c} \times 100 \quad (6)$$

$$\text{F1-score}_c = 2 \times \frac{(\text{Precision}_c \times \text{Recall}_c)}{\text{Precision}_c + \text{Recall}_c} \quad (7)$$

4.4. Experimental Result

Table 3 shows the classification success rate, the completely matched rate and the average rank of each classifier. “Char. freq.” indicates a classifier which is trained by MLP using character frequency vectors. “CF-IAF” indicates a classifier which is trained by CNN using character importance matrix as features. “Image” indicates a classifier which is trained by CNN using AA which is converted into edge images. “Char. image” indicates a classifier which is trained by MLP using character’s HOG image class vector.

Table 3. Experimental Results

	Char. freq.	CF-IAF	Image	Image (VggNet)	Char. image
Success rate (%) [k=3]	41.19	21.61	32.96	37.95	45.22
Success rate (%) [k=5]	49.57	28.92	39.18	48.22	52.65
Success rate (%) [k=10]	64.30	44.16	51.72	64.10	65.96
Complete match rate (%)	27.59	11.12	25.64	21.37	31.97
Average of rank	3.31	4.31	3.30	3.60	3.05
Macro-averaged Precision (%)	28.74	9.38	25.53	21.89	29.83
Macro-averaged Recall (%)	31.41	13.13	45.50	33.08	35.33
Macro-averaged F1-score	26.87	9.41	28.51	21.24	29.62

As the experimental results, the method based on character HOG class vector obtained the highest classification success rate, the highest complete match rate and the highest averaged rank. The method based on image feature obtained the highest macro-averaged recall for each category.

4.5. Cross Validation Test

We conducted a 5-fold cross validation test for five types of methods as described above. Table 4 shows the number of the experimental data. The experimental results are shown in Table 5. All evaluation scores by “Char. image” are better than the evaluation scores of the other methods.

Table 4. Experimental Data (5-fold Cross Validation)

Training Data	Validation Data	Test Data
38278 × 5	9570 × 5	11962 × 5

Table 5. Experimental Results (5-fold Cross Validation)

	Char. freq.	CF-IAF	Image	Image (VggNet)	Char. image
Success rate (%) [k=3]	11.47	11.93	29.87	23.73	52.83
Success rate (%) [k=5]	18.58	18.64	36.44	32.16	60.26
Success rate (%) [k=10]	33.93	34.57	49.93	48.14	71.82
Complete match rate (%)	4.05	4.47	21.25	12.63	38.70
Average of rank	5.20	5.17	3.58	4.22	2.73
Macro-averaged Precision (%)	1.98	0.38	23.15	15.52	39.37
Macro-averaged Recall (%)	2.18	2.29	22.03	11.75	39.11
Macro-averaged F1-score	0.65	0.23	22.49	11.78	39.18

Fig. 8 shows the Precision, Recall and F1-score for each category by the method using “Char. image feature.” From the results, it was found that some categories could obtain over 50% precision. On the other hand, the categories with low precision are also over 20%. The difference between the maximum precision and the minimum precision is approximately 50%. As the reason that there are difference in the classification difficulty for each category, it is considered that whether the similar categories are included or not, and whether the similar ASCII arts are included in the same category.

5. Discussions

This section discusses the differences of each method, referring to the examples of the higher ranked accuracy categories and the lower ranked accuracy categories. Table 6, 7, 8, 9 show the categories ranked in the top 10 of accuracy and the worst 10 of accuracy in each method. When we look at only the top 10 categories, the method using image feature achieved over 70% classification accuracy for all categories. On the other hand, the method using character importance value could not obtain over 60% accuracy for each category.

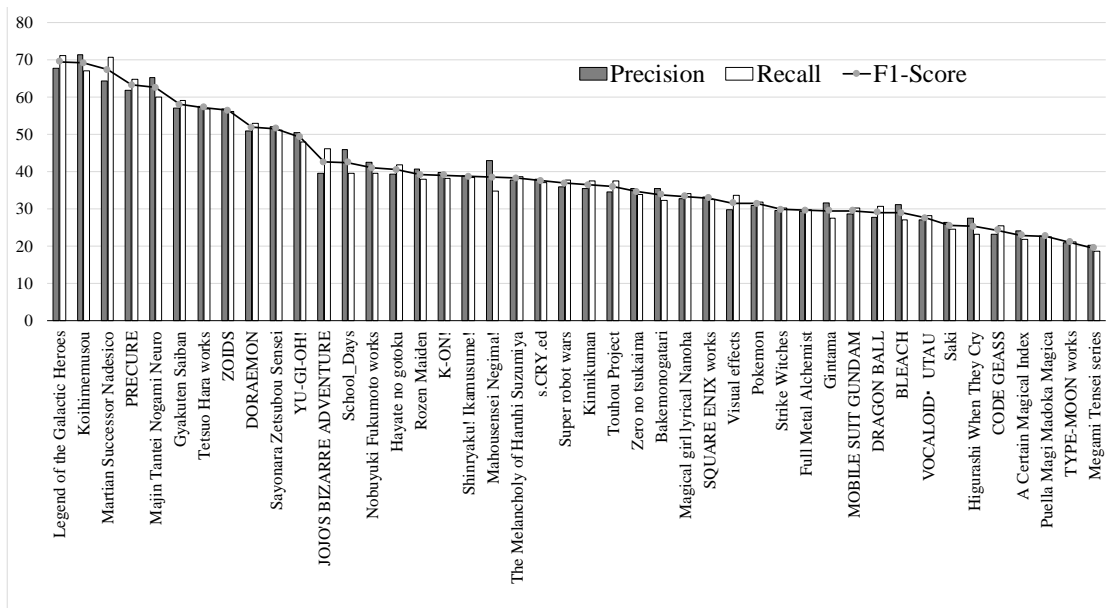


Fig. 8. Precision, Recall, F1-scores for each category in 5-fold cross validation test (Char. image feature).

Table 6. Accuracy for Each Category of the Method Using Frequency of Character

Rank	Category	Accuracy	Rank	Category	Accuracy
1	Martian Successor Nadesico	83.59	35	Rozen Maiden	8.29
2	Legend of the Galactic Heroes	73.36	36	CODE GEASS	6.51
3	Majin Tantei Nogami Neuro	66.67	37	Gintama	6.45
4	Koihimemusou	65.91	38	Strike Witches	5.57
5	PRECURE	64.25	39	BLEACH	5.26
6	Tetsuo Hara works	62.99	40	VOCALOID · UTAU	3.44
7	Sayonara Zetsubou Sensei	56.59	41	MOBILE SUIT GUNDAM	3.43
8	ZOIDS	55.36	42	Megami Tensei Series	3.42
9	JOJO'S BIZARRE ADVENTURE	51.3	43	TYPE-MOON works	2.96
10	Nobuyuki Fukumoto Works	47.11	44	Puella Magi Madoka Magica	2.27

Table 7. Accuracy for Each Category of the Method Using Importance Value of Character

Rank	Category Name	Accuracy	Rank	Category Name	Accuracy
1	Koihimemusou	52.27	35	CODE GEASS	6.13
2	Gyakuten Saiban	42.86	36	Magical girl lyrical Nanoha	6.09
3	PRECURE	30.33	37	Rozen Maiden	5.53
4	Tetsuo Hara works	29.92	38	The Melancholy of Haruhi Suzumiya	5.29
5	DORAEMON	29.5	39	Saki	4.86
6	Legend of the Galactic Heroes	27.1	40	Touhou Project	4.68
7	Pokemon	22.23	41	MOBILE SUIT GUNDAM	3.84
8	Majin Tantei Nogami Neuro	21.97	42	visual effects	3.75
9	Martian Successor Nadesico	17.56	43	Puella Magi Madoka Magica	3.54
10	s.CRY.ed	17.39	44	YU-GI-OH!	3.07

Table 8. Accuracy for Each Category of the Method Using Image Feature

Rank	Category Name	Accuracy	Rank	Category Name	Accuracy
1	BLEACH	92.63	35	Magical girl lyrical Nanoha	17.34
2	Gyakuten Saiban	91.84	36	MOBILE SUIT GUNDAM	17.15
3	Gintama	87.1	37	Puella Magi Madoka Magica	17.14
4	Bakemonogatari	80.81	38	YU-GI-OH!	16.67
5	Kinnikuman	80.73	39	visual effects	15.27

6	Tetsuo Hara works	74.8	40	PRECURE	14.37
7	Koihimemusou	73.3	41	TYPE-MOON works	13.46
8	Sayonara Zetsubou Sensei	71.32	42	The Melancholy of Haruhi Suzumiya	13.25
9	Legend of the Galactic Heroes	70.56	43	Touhou Project	12.86
10	s.CRY.ed	70.29	44	Rozen Maiden	11.91

Table 9. Accuracy for Each Category of the Method Using Character Image Feature

Rank	Category Name	Accuracy	Rank	Category Name	Accuracy
1	Martian Successor Nadesico	83.97	35	TYPE-MOON works	19.31
2	Legend of the Galactic Heroes	71.03	36	A Certain Magical Index	18
3	PRECURE	70.87	37	Strike Witches	16.04
4	Majin Tantei Nogami Neuro	68.94	38	BLEACH	15.79
5	Koihimemusou	67.05	39	Bakemonogatari	12.12
6	Tetsuo Hara works	66.14	40	Rozen Maiden	11.78
7	Sayonara Zetsubou Sensei	64.34	41	VOCALOID・UTAU	10.64
8	ZOIDS	63.1	42	CODE GEASS	9.58
9	DORAEMON	50.36	43	Megami Tensei series	4.79
10	JOJO'S BIZARRE ADVENTURE	48.1	44	Puella Magi Madoka Magica	4.25

Fig. 9 shows the ASCII art in the category of “BLEACH” whose accuracy was the highest in all of the five methods. This image gives us an impression that it is rather a complexed picture. Among the other ASCII arts belonging to this category, there were a lot of ASCII arts that were almost the same and had small differences. The total number of the ASCII arts under this category is 913, and we used all of the data for training and evaluation. Therefore, the result seems to be much influenced by the fact that a lot of similar ASCII arts were included in the same category rather than the fact that the ASCII arts had distinctive design or style.

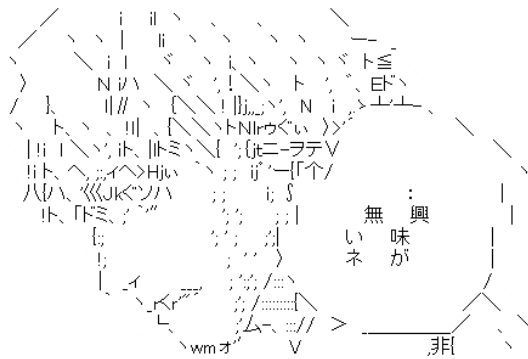


Fig. 9. ASCII art image of category “BLEACH”.

However, the method using image feature showed different tendency compared to other methods using character frequency or character importance. Concretely, the categories obtained higher accuracy are different from those of other methods. Thus, we can say that the method using image feature did not receive so much influence from the kind of the characters. Therefore, the method using image feature has high general versatility.

The categories of “Tetsuo Hara works” and “Legend of the Galactic Heroes” achieved high accuracy in common in the five methods. When we compared the number of the subcategories belonging to each category, “Tetsuo Hara works” had 19 kinds and “Legend of the Galactic Heroes” had 24 kinds, which means there were no large difference.

Moreover, there were such differences as in facial expressions only, entire body or only head, existence of scripts and the number of the characters drawn in one ASCII art (one character or more than one characters). They share the point that a lot of similar ASCII arts belong to the same categories

In any case, even though the sizes of ASCII arts are different from each other, the kinds of the characters used in the ASCII arts are the same. So, it seems that the method using character feature could achieve high accuracy.

On the other hand, in the categories with the lower accuracy, the method using image feature and the method using character image vector obtained more than double accuracy compared to other two methods. This means that these methods could capture the characteristics of the pictures under the categories by using the shape features obtained from the image features. One of the reasons why the total accuracy was low is because many similar ASCII arts were included in the largest category in the evaluation data. In the similar ASCII arts the same characters were used repeatedly. Therefore, the character frequency feature would have become an advantage. The category of “Mobile Suit Gundam” ranked in the lowest 10. Because the “mobile suit” and “human” were included in the same category, the ASCII arts in this category might have been difficult to be classified into the same category.

Some data such as ASCII arts are comparatively difficult to be collected. To use such data for training, it would be necessary to augment the data by adding artificial noise or to classify the categories by predicting the source images from the line arts.

Fig. 10 shows the confusion matrix among the top 10 ranked categories with the lowest accuracy method using character importance value. Fig. 11 shows the confusion matrix among the top 10 ranked categories with the method using image feature that resulted in the highest average accuracy for each category.

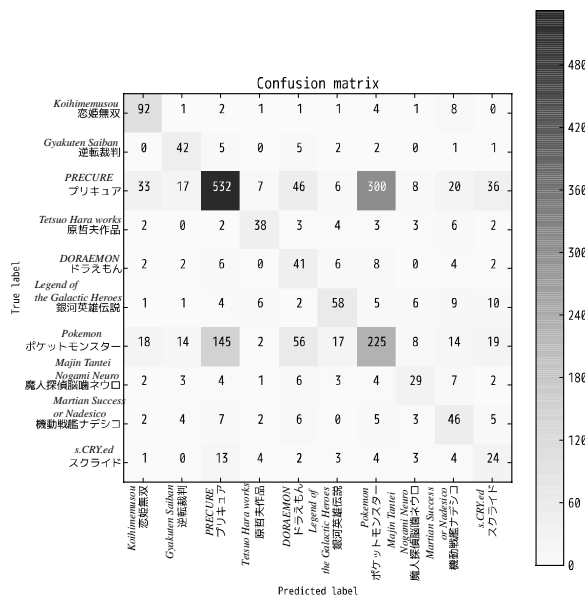


Fig. 10. Confusion matrix of the top 10 categories (CF-IAF).

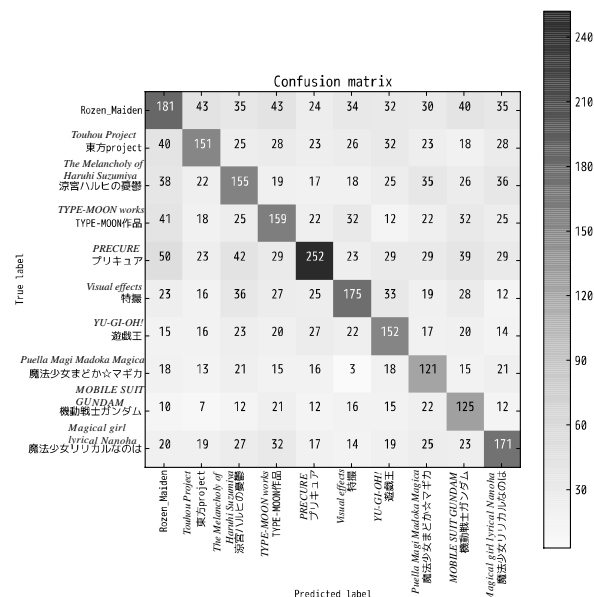
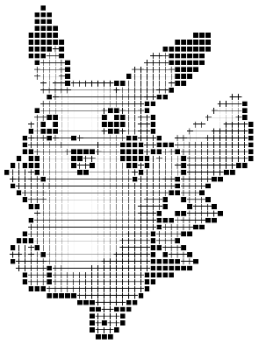


Fig. 11. Confusion matrix of lower 10 ranked categories (Image feature).

As is the figure, a lot of classification errors occurred in the method using character importance value. Especially, the categories of “Pokémon,” “PRECURE” and “DORAEMON” were often classified wrongly. Because all of these works are anime works for kids, we think that the picture of their ASCII arts were similar each other. On the other hand, there were not many bias of the misclassification tendency in the method using image feature. The ASCII arts were rather easily classified into every category to the same degree.

5.1. Success Example

In this subsection, we introduce the success example by the proposed method. Fig. 12 shows the examples of ASCII arts which were successfully classified into correct category in the evaluation experiment.



Category: Pokemon

This ASCII art is created with dot image format (tone-based ASCII art).



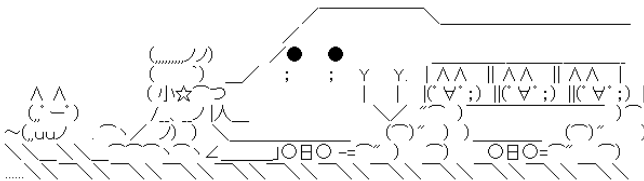
Category: The Melancholy of Haruhi Suzumiya

This ASCII art is silhouette type. The silhouette type ASCII art is rare in the other category.

Fig. 12. Success examples.

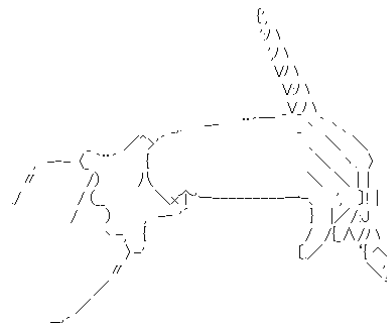
5.2. Failure Example

In this subsection, we introduce the failure example by the proposed method. Fig. 13 shows the examples of ASCII arts which were misclassified in the evaluation experiment.



Category: Kinnikuman

This ASCII art is extremely deformed.



Category: PRECURE

Because this is a part of body ASCII art, characteristic of category cannot be expressed well.

Fig. 13. Failure examples.

6. Conclusion

In this paper, we compared and considered the method based on character feature and the method based on image feature to classify ASCII arts. As the categories of the ASCII arts, we set the title or series title of the ASCII arts in motif of comics, anime, and game works, and trained the classifier by neural networks.

In the experimental results, the method achieved the highest accuracy when the classifier made by training the character image vectors of the character-level HOG feature by MLP was used. The method achieved the highest accuracy (Maximum 92.63%) for each category when the classifier trained image feature by CNN was used.

As future tasks, by conducting evaluation experiment, we would like to validate the effectiveness of image features in the case where the method using character frequent vector is thought to be disadvantageous.

That is the case where the different characters are used to express the same object. Besides, not only classifying by works, we would like to consider classifying whether “deform” or “not deform” and recognizing the categories of the objects that are motif of the ASCII arts.

We are also planning to convert pictures or illustration images into edge images, to create an ASCII art category classifier by using non-ASCII arts as training data and to evaluate the classifier.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP15K16077, JP18K11549.

References

- [1] Tanioka, H., & Maruyama, M. (2005). Ascii art pattern recognition using SVM based on morphological analysis. *The Institute of Electronics, Information and Communication Engineers*, 104(668), 25-30.
- [2] Hayashi, K., & Suzuki, T. (2009). A language independent text art extraction method based on text compression. *Technical Report Digital Document (DD)*.
- [3] Suzuki, T. (2011). Text normalization on the text art extraction method using data compression rate. *Proceeding of the 17th of The Annual Meeting of the Association for Natural Language Processing*.
- [4] Yamada, T., Tsuchiya, S., Kuroiwa, S., & Ren, F. (2007). Classification of facemarks using n-gram. *Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering*.
- [5] Matsumoto, K., Fujisawa, A., Yoshida, M., & Kita, K. (2017). Emotion recognition of emoticons based on character embedding. *Journal of Software*, 12(11), 849-857.
- [6] Fujisawa, A., Matsumoto, K., Yoshida, M., & Kita, K. (2017). Facial expression classification based on shape feature of emoticons. *Proceedings of 1st International Conference on Machine Learning and Data Engineering*, 29-34.
- [7] Jirela, D., Matsumoto, K., Yoshida, M., & Kita, K. (2018). Predicting facial expression from emoticon based on convolutional neural networks. *Proceedings of the 4th International Forum on Advanced Technologies (IFAT)*.
- [8] Fujisawa, A., Matsumoto, K., Yoshida, M., & Kita, K. (2016). Detecting emoticons in ASCII arts using image features. *Proceedings of the 30th Annual Conference of the Japanese Society for Artificial Intelligence*.
- [9] Fujisawa, A., Matsumoto, K., Yoshida, M., & Kita, K. (2017). Classification method for large ASCII arts using image features. *Proceedings of the 31th Annual Conference of the Japanese Society for Artificial Intelligence*.
- [10] Xu, X., Zhang, L., & T. Wong. (2010). Structure-based ASCII Art. *ACM Transactions on Graphics*, 29(4).
- [11] Takeuchi, Y., Takafuji, D., Ito, Y., & Nakano, K. (2014). ASCII art generation using the local exhaustive search on the GPU. *Proceedings of the First International Symposium on Computing and Networking*.
- [12] DeepAA on the Web Version 2.1. Retrieved from <https://tar-bin.github.io/DeepAAonWeb/>
- [13] AAtoImage. Retrieved from <http://www.nicovideo.jp/watch/sm20296302>
- [14] Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 886-893.
- [15] Zhao, Y. & Karypis, G. (2002). Comparison of agglomerative and partitional document clustering algorithms. *Technical report, Department of Computer Science*.
- [16] Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5), 513-523.
- [17] Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. Retrieved from <http://arxiv.org/abs/1409.1556v6>.

[18] Keras. Retrieved from <https://keras.io/>

[19] TensorFlow. Retrieved from <https://www.tensorflow.org/>



Kazuyuki Matsumoto received the PhD degree in 2008 from Tokushima University. He is currently an assistant professor of Tokushima University. His research interests include affective computing, emotion recognition, artificial intelligence and natural language processing. He is a member of IPSJ, ANLP, IEICE, JSAI, and IEEJ.



Akira Fujisawa received the PhD degree in 2018 from Tokushima University. He is currently an assistant professor of Aomori University. His research interests include multimedia information retrieval, image recognition. He is a member of JSAI.



Minoru Yoshida is a lecturer at the Department of Information Science and Intelligent Systems, University of Tokushima. After receiving his BSc, MSc, and PhD degrees from the University of Tokyo in 1998, 2000, and 2003, respectively, he worked as an assistant professor at the Information Technology Center, University of Tokyo. His current research interests include web document analysis and text mining for the Documents on the WWW.



Kenji Kita received the B.S. degree in mathematics and the PhD degree in electrical engineering, both from Waseda University, Tokyo, Japan, in 1981 and 1992, respectively. From 1983 to 1987, he worked for the Oki Electric Industry Co. Ltd., Tokyo, Japan. From 1987 to 1992, he was a researcher at ATR Interpreting Telephony Research Laboratories, Kyoto, Japan. Since 1992, he has been with Tokushima University, Tokushima, Japan, where he is currently a professor at Faculty of Engineering. His current research interests include multimedia information retrieval, natural language processing, and

speech recognition.