# Utilizing Visual Forms of Japanese Characters for Neural Review Classification

**Yota Toyama** **Makoto Miwa** **Yutaka Sasaki**
Toyota Technological Institute
2-12-1 Hisakata, Tempaku-ku, Nagoya, Aichi, 468-8511, Japan
{sd16423, miwa-makoto, yutaka.sasaki}@toyota-ti.ac.jp

## Abstract

We propose a novel method that exploits visual information of ideograms and logograms in analyzing Japanese review documents. Our method first converts font images of Japanese characters into character embeddings using convolutional neural networks. It then constructs document embeddings from the character embeddings based on Hierarchical Attention Networks, which represent the documents based on attention mechanisms from a character level to a sentence level. The document embeddings are finally used to predict the labels of documents. Our method provides a way to exploit visual features of characters in languages with ideograms and logograms. In the experiments, our method achieved an accuracy comparable to a character embedding-based model while our method has much fewer parameters since it does not need to keep embeddings of thousands of characters.

## 1 Introduction

Some languages like Japanese and Chinese have ideograms and logograms that are characters representing words or phrases by themselves. In these languages, such kinds of characters usually have the same visual (surface) components (radicals) when they have similar semantic or phonetic features. Figure 1 illustrates three Japanese *Kanji* characters related to fish. These Kanji characters share the same visual components unlike English characters or words. This kind of shared components often appears in the Kanji characters as shown in Table 1. Most natural languages methods, however, ignore the visual information since they often treat texts as sequences of symbolic values like integer indices. They therefore lose significant useful information in processing texts in such languages. Furthermore, these languages usually contain many kinds of characters. For example, a typical Japanese character set JIS X 0213 contains 11,233 different characters including Hiraganas, Katakanas, and Kanjis. This large number of characters often makes it difficult to apply recent character embedding models to such languages, and this fact prompts us to reduce the number of parameters used to store information for characters.

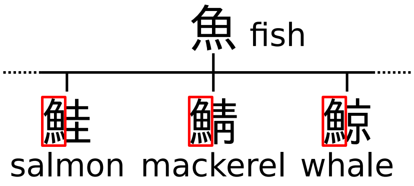

Figure 1: Kanji characters relevant to fish and their sharing components (radicals)

| Component | Kanji characters |
|---|---|
| 食 (eat) | 飯 (food), 飲 (drink), 餐 (meal) |
| 土 (soil) | 地 (earth), 場 (field), 坂 (slope) |

Table 1: Kanji characters with shared components

We propose a novel method to analyze Japanese review documents with exploiting the visual information of ideograms and logograms. Our method extends character-based Hierarchical Attention Networks (HAN) (Yang et al., 2016) by incorporating visual information of characters. The method first builds character embeddings from their font images and then feeds them as inputs into the character-based HAN.

Our main contribution is to show the usability of font images as potential character representation not to use them as additional information but to substitute for integer indices. Our method represents documents without the need for external

*Proceedings of the The 8th International Joint Conference on Natural Language Processing*, pages 378–382,
Taipei, Taiwan, November 27 – December 1, 2017 ©2017 AFNLP

character dictionaries like radical dictionaries and without depending on the characters such as Hiraganas, Katakanas, and Kanjis. Additionally, we show our method can simplify a baseline model with reducing the number of parameters by adopting a convolutional neural network (CNN) to extract character features from font images.

## 2 Baseline model

Our method is based on a review classification model named Hierarchical Attention Networks (HAN) (Yang et al., 2016). We employ this method since this is one of the state-of-the-art methods in sentiment classification on English datasets of Yelp, IMDB, Yahoo Answer, and Amazon reviews, and we aim to evaluate the visual information on the state-of-the-art model. The HAN model is composed of bidirectional Recurrent Neural Networks of Gated Recurrent Units (GRU-RNNs). The RNNs are stacked hierarchically from a word level to a sentence level. The model encodes a sequence of lower-level embeddings to an upper-level embedding in a bottom up manner with attention mechanisms. For example, a sentence embedding is calculated from the word embeddings in a sentence, and a document embedding is calculated from the sentence embeddings in a document. The attentions are calculated using the outputs of lower-level RNNs and then applied to the outputs to calculate the embedding of each upper-level element as follows:

$$\mathbf{h}_i = \tanh(W_w \mathbf{l}_i + \mathbf{b}_w)$$

$$\mathbf{u} = \sum_i \alpha_i \mathbf{h}_i, \; \alpha_i = \frac{\exp(\mathbf{h}_i^\top \mathbf{c})}{\sum_i \exp(\mathbf{h}_i^\top \mathbf{c})} \qquad (1)$$

where $\mathbf{l}_i$ is the embedding of a $i$-th lower-level element in a sequence and $\mathbf{u}$ is the embedding of an upper-level element. $W_w$, $\mathbf{b}_w$ and $\mathbf{c}$ are parameters to be tuned during training. $\mathbf{c}$ also provides a way to investigate the grounds of predictions with attention mechanisms. This hierarchical architecture allows to suppress the effects of gradient vanishing when RNNs are applied to long sequences.

## 3 Proposed method

We propose a novel model that utilizes visual font image information of characters to represent characters. Using font images for Japanese has several merits compared with symbolic features of characters. First, font images are available to any characters unlike some character-specific features that need to be treated differently. For instance, radicals are specific to Kanjis, and dictionaries are required to extract such features. Second, we should be able to find proper font images for characters in some way most of the time. That is because most datasets are composed of documents written on some computers and people should not use characters that do not have proper font images as they cannot be rendered on their systems. Third, we can reduce the number of parameters.

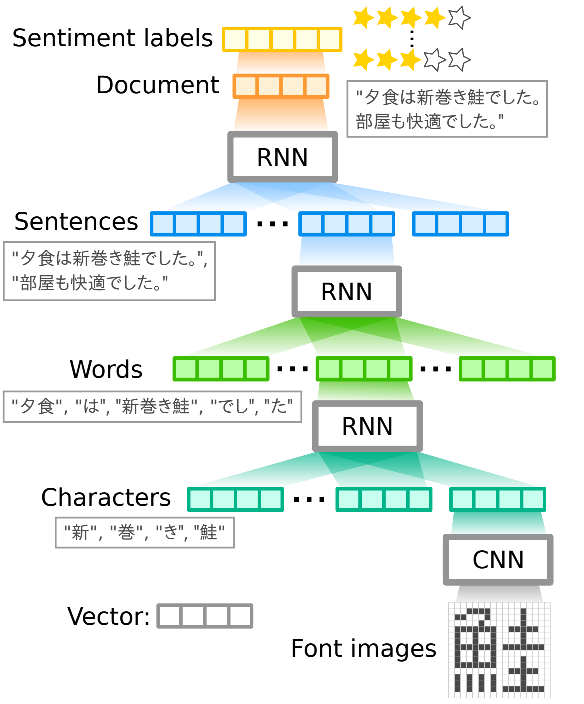

Figure 2: Proposed model for a review document “夕食は新巻き鮭でした。部屋も快適でした。” “The dinner was a lightly salted salmon. The room was comfortable too.”

We convert each font image into its corresponding character embedding with CNN and incorporate them into a character-based HAN model, which is a straight-forward extension of the HAN model with character-level RNNs[1].

Font images are extracted as fixed-size images, and they are used statically throughout training and evaluation. All the pixel values in the font images, which are originally with a range of $[0, 255)$ of integers, are normalized into real values with a range of $[0, 1)$. The CNN consists of five convolutional layers interleaved by pooling layers[2]. We

[1] Yang et al. (2016) mentioned the possibility of character-based HAN in their paper, but they did not evaluate it.

[2] We empirically chose this number of layers. We also tried to use VGG (Simonyan and Zisserman, 2015) with Xavier initialization (Glorot and Bengio, 2010) to convert

stack the character-based HAN on the top of the CNN to hierarchically structure documents from characters to words, sentences, and documents. The final document embeddings are passed to softmax functions through a hidden layer to predict sentiment labels. Since our task consists of multiple classification tasks, we prepare an individual softmax function for each classification task in the output layer with sharing the hidden layer.

## 4 Experiments

### 4.1 Dataset

We used 320,000 reviews in the dataset of Rakuten Travel review[3]. We split them into training, development, and test data with 300,000, 10,000, and 10,000 reviews respectively. The task is multi-category sentiment classification; the task consists of 6-class (0 (no rating) and 1 (bad) to 5 (good)) sentiment classifications for seven categories (location, room, food, bath, service, facility, and overall). All documents were normalized by NFKC Unicode normalization and then by a Japanese text normalizer `neologdn`[4]. The documents were segmented into sentences by a regular expression, and every sentence was segmented into words by a Japanese morphological analyzer MeCab (Kudo et al., 2004).

Character and word vocabularies were constructed from those appeared more than nine times in training and development datasets[5]. As a result, we chose 2,709 characters from 3,630 characters. We employed the IPA Gothic TrueType font[6] to represent characters.

### 4.2 Experimental settings

We compared our font-based model with the character-based HAN. We used Python and TensorFlow to implement the models, and ran them on an NVidia GeForce GTX TITAN X. We reimplemented the HAN model from scratch and extended it to implement our model.

We optimized all the models by Adam with suggested parameters on the paper (Kingma and Ba, 2015). We employed mini-batch training and batch sizes were fixed to 16. For the character-based HAN, character, word, sentence and document embedding sizes were set to 100, 150, 100 and 50 respectively. Note that the embedding sizes for the inputs of upper layer (RNN or hidden layer) were doubled before they were fed to the upper layer since GRU-RNNs were bidirectional and embeddings of outputs from lower forward and backward RNNs were concatenated. For example, the size of the last hidden layer was set to 100. We tuned the other hyper-parameters in a greedy strategy. As for regularization, we applied L2 regularization with a scale of 1e-8 and did not use dropout. The model was updated in 66,348 times.

As for our font-based model, the font images were represented as 2-dimensional matrices of 64×64 single-channel images with 8-bit depth. Each hidden image from each pooling layer in the CNN part of our model had 32 channels. The resulting hidden images were 2x2 32-channel images, which were then flatten as character embeddings before they were fed to the RNN that converted character embeddings into word embeddings. The sizes of word, sentence, and document embeddings were set to 150, 100 and 50 respectively. We updated the model in 206,230 times from scratch without any regularization or any pretraining of the CNN.

### 4.3 Results

We show the numbers of parameters for character embeddings in Table 2. This table shows that our model needs less parameters than the character embedding-based model since the number of parameters in our model does not depend on the number of character types. This table indicates that character-based HAN can keep only 374 characters with the similar model size to ours.

The accuracies of the models on the Rakuten Travel dataset are shown in Table 3. We show the results with an embeddings-based classification method by Toyama et al. (2016) for reference. As the table shows, the plain HAN works better than the existing method and our method achieved an accuracy comparable to a plain HAN. The result indicates two insights. First, our model extracted character features successfully from font images in spite of the complexity of images, deep CNN architecture and less parameters, and the font images can be an alternative for symbolic character

font images without pre-training, but it did not work well even after many training epochs.

[3] http://www.nii.ac.jp/dsc/idr/en/rakuten/rakuten.html

[4] https://github.com/ikegami-yukino/neologdn

[5] We empirically chose this threshold. We got lower score when we used all the characters.

[6] http://ipafont.ipa.go.jp/

| Method | #parameters |
|---|---|
| character-based HAN | 270,900 |
| Our method | 37,312 |

Table 2: Comparison of the numbers of parameters related to character embeddings

| Method | Accuracy (%) |
|---|---|
| Toyama et al. (2016) | 50.2 |
| character-based HAN | 53.4 |
| Our method | 53.3 |

Table 3: Accuracies of methods on the Rakuten Travel dataset

indices in representing characters. Second, the use of font images to represent characters is reasonable for the multi-category sentiment classification.

## 5 Related work

Many deep learning models have been proposed for sentiment classification and have achieved the state-of-the-art performance. These models grasp and utilize dynamics in natural languages, such as negation and emphasis relations among words and sentences. Yang et al. (2016) proposed Hierarchical Attention Networks (HAN), which are composed of hierarchically stacked RNNs, and each RNN captures dynamics of words or sentences. Our model extends this model by incorporating visual information of characters.

Some shallow models are still comparable with the deep learning models. FastText (Joulin et al., 2016) employs a multi-layer perceptron, which constructs a hidden document embedding from unigram and bigram embeddings and classifies the document using the document embedding. Our CNN model can be used to incorporate visual features of characters into these models.

The most similar work to ours is the work by Costa-juss et al. (2017) since they used font images in their method, although their target task is not review classification but neural machine translation. They initialized embeddings with bitmap fonts, and they achieved a better BLEU score than a baseline method without bitmap fonts of Chinese characters. They, however, did not directly incorporated the font images into their models unlike ours and they used the font information as additional information, so the parameters were increased by using font images in their model.

Several other related work has exploited processing the character components, mostly radicals, in Japanese (Yencken and Baldwin, 2008) and Chinese (Jin et al., 2012; Lepage, 2014; Shi et al., 2015; Li et al., 2015; Dong et al., 2016). Sun et al. (2014) proposed radical-enhanced Chinese character embeddings for word segmentation in Chinese. They utilized radical information of Chinese characters using a radical-mapping dictionary. Their model consists of two models for words segmentation and radical prediction with sharing parameters of character embeddings. They incorporate the radical information into character embeddings by this radical prediction. Their method was tailored for Chinese where all the characters have radicals as character components. Some kinds of Japanese characters like Hiraganas and Katakanas are syllabograms that do not represent words, so their method is not directly applicable to Japanese. Also, most of the existing work depends on dictionaries. Our method models the visual character information directly, so our method is applicable to Chinese or any other languages without any dictionary.

## 6 Conclusion

We proposed a method for a multi-category sentiment classification that exploits font images as potential representation of documents. The experimental results showed that our method performs as well as the plain character-based HAN on a dataset of Rakuten Travel reviews with reducing the number of parameters. The results suggest that our method can utilize visual features of font images successfully to represent characters and such visual information works well for multi-category sentiment classification.

As future work, we would like to investigate the better modeling of the font images by incorporating an attention mechanism to represent the locations of font images. This will enable us to investigate how our model works on the task by checking whether the visual attentions are paid on character components like radicals in Kanji characters. We would also like to compare and/or combine our method with its variants with more symbolic character features like radial information from Kanji dictionaries. That should help existing methods to run on test datasets with the existence of unknown characters since our method does depend not on artificial hand-crafted features of characters extracted from dictionaries that may lack some rare characters but only on visual information of characters.

## Acknowledgments

This work was supported by the TOYOAKI Scholarship Foundation.

## References

Marta R. Costa-juss, David Aldn, and Jos A. R. Fonollosa. 2017. Chinese − spanish neural machine translation enhanced with character and word bitmap fonts. In *Machine Translation*, page 1 − 13, Barcelona, Spain.

Chuanhai Dong, Jiajun Zhang, Chengqing Zong, Masanori Hattori, and Hui Di. 2016. Character-based LSTM-CRF with radical-level features for chinese named entity recognition. In *The Fifth Conference on Natural Language Processing and Chinese Computing & The Twenty Fourth International Conference on Computer Processing of Oriental Languages(NLPCC-ICCPOL 2016)*, pages 239–250, Kunming, China.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS' 10)*. Society for Artificial Intelligence and Statistics.

Peng Jin, John Carroll, Yunfang Wu, and Diana McCarthy. 2012. Distributional similarity for chinese: Exploiting characters and radicals. *Mathematical Problems in Engineering*, 2012.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint*, abs/1607.01759.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR2015)*, San Diego, CA.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying conditional random fields to japanese morphological analysis. In *Proceedings of EMNLP 2004*, pages 230–237, Barcelona, Spain. ACL.

Yves Lepage. 2014. Analogies between binary images: Application to chinese characters. In *Computational Approaches to Analogical Reasoning: Current Trends*, pages 25–57. Springer.

Yanran Li, Wenjie Li, Fei Sun, and Sujian Li. 2015. Component-enhanced chinese character embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 829–834, Lisbon, Portugal. ACL.

Xinlei Shi, Junjie Zhai, Xudong Yang, Zehua Xie, and Chao Liu. 2015. Radical embedding: Delving deeper to chinese radicals. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 594–598, Beijing, China. Association for Computational Linguistics.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR2015)*, San Diego, CA.

Yaming Sun, Lei Lin, Nan Yang, Zhenzhou Ji, and Xiaolong Wang. 2014. Radical-enhanced chinese character embedding. In *International Conference on Neural Information Processing*, pages 279–286. Springer.

Yota Toyama, Makoto Miwa, and Yutaka Sasaki. 2016. Rating prediction by considering relations among documents and sentences and among categories. In *Proceedings of the Twenty-second Annual Meeting of the Association for Natural Language Processing*, pages 158–161. (In Japanese).

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of NAACL-HLT*, pages 1480–1489, San Diego, CA. ACL.

Lars Yencken and Timothy Baldwin. 2008. Measuring and predicting orthographic associations: Modelling the similarity of japanese kanji. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 1041–1048, Manchester, UK. Coling 2008 Organizing Committee.