16th International Conference on Machine Vision Applications (MVA)
National Olympics Memorial Youth Center, Tokyo, Japan, May 27-31, 2019.

News2meme: An Automatic Content Generator from News Based on Word Subspaces from Text and Image

Erica K. Shimomoto, Lincon S. Souza
University of Tsukuba
Tsukuba, Japan
erica,lincons{@cvlab.cs.tsukuba.ac.jp}

Bernardo B. Gatto, Kazuhiro Fukui
Center for Artificial Intelligence Research
Tsukuba, Japan
bernardo@cvlab,kfukui@{cs.tsukuba.ac.jp}

Abstract

Internet users engage in content creation by using various media formats. One of the most popular forms is the "internet meme", which often depicts the general opinion about events with an image and a catchphrase. In this paper, we propose news2meme, a method for automatically generating memes from a news article, where we aim to match words and images efficiently. We approach this as two multimedia retrieval problems with the same input news text: 1) An image retrieval task where the output is a meme image; 2) A text retrieval task where the output is a catchphrase. These two outputs are combined to generate the meme for the news article. We represent texts and catchphrases as sets of word vectors through the word2vec representation. To handle images similarly, we extract sets of tags from the images using a deep neural network. These tags are then translated to word vectors in the same vector space through word2vec. Finally, we represent the intrinsic variability of features in a set of word vectors with a word subspace. Through word subspaces comparison, we can directly compare images and texts, making retrieval across media formats possible. A preliminary experiment was performed to evaluate our framework.

1 Introduction

Content creation is one of the main activities on social media networks. Among this content, memes have gained prominence due to their simplicity and comicality. This term is used to describe an activity, concept, catchphrase, or piece of media (e.g., an image, hyperlink, video, website, or hashtag) that spreads by mimicry or for humorous purposes via the Internet [1]. One of the most popular types of memes is the "image macro" meme [2], a combination of a phrase and image, which uses irony and sarcasm to depict a general opinion about recent events.

Previous works have successfully generated memes from posts on Twitter and news headlines; however, most of them did not extract information automatically from both images and texts to create the memes. Costa et al. [3] search for the most frequent nouns associated with a public figure and replaces them in quotes, creating new phrases. The image is, however, retrieved from the internet using a search engine, with no analysis on the image context. In [4], despite using common meme images, the headlines matching is based on a set of rules manually defined for each image.

Wang and Wen [5] studied the correlation among popular meme images and their wordings, retrieving meme descriptions from raw images. Their results showed that extracting information from both image and text generates meaningful memes, with descriptions more coherent with the image context.

Considering the above discussion, we propose *news2meme*, a framework for automatic "image macro" meme generation. Our input is a news text, and the output is the meme, which is composed by an image and a catchphrase. Our problem is then formulated as two multimedia retrieval tasks where we wish to retrieve a meme image and a catchphrase that match well the content of the input news text.

To solve our problem, we need to compare and match three different types of information sources: a meme image, a catchphrase, and a news text. To this end, they are required to be represented in a common form for the direct comparison. Our basic idea to address this issue is to represent the three sources as sets of word vectors as follows: Words in the news text and catchphrase are translated to word vectors using the *word2vec* representation [6]. For the images, first, a set of tags is extracted from them by using a deep neural network. These tags are then translated to word vectors by using the *word2vec*.

Under this framework, we represent each set of words compactly as a word subspace [7] in the same vector space. The word subspace is mathematically defined as a low-dimensional linear subspace in a high-dimensional vector space. We can calculate the similarity between two word subspaces by using the mu-

tual subspace method (MSM) [8]. Thus, we can link and compare the different types of information sources naturally and effectively. In this way, we can realize the framework to retrieve the meme image and catchphrase from a given news text query.

The rest of this paper is organized as follows. In Section 2, we describe our proposed meme generator, explaining how to match the three types of media. Tests performed to evaluate our framework, and their main results are described in Section 3. Finally, our main conclusions are presented in Section 4.

2 Proposed Meme Generator

In this section, first, we give a general overview of our framework with the basic concept behind it. Then, we explain how to model a word subspace from texts and images, and how retrieval is performed through word subspace.

2.1 Framework overview

The primary goal of *news2meme* is to generate a meme from a news text. Figure 1 shows our framework. To generate a meme, we find an image and a catchphrase correspondent to the news text through the comparison of word subspaces.

Our framework has two different stages: A learning and a generation stage. In the learning stage, we consider two different sources: $\mathcal{S}_{img}$, with meme images; and $\mathcal{S}_{cphr}$, with catchphrases. For each meme image and each catchphrase in those sources, we model a word subspace, resulting in sets of catchphrase word subspaces, $\mathcal{Y}_{cphr}$, and sets of image word subspaces, $\mathcal{Y}_{img}$. These are the reference word subspaces.

Then, in the generation stage, for a given input news text in the source $\mathcal{S}_{txt}$, we model an input word subspace $\mathcal{Y}_{txt}$. We calculate the similarity between the input word subspace and the reference word subspaces (images and catchphrases) using MSM. The image and catchphrase with the highest similarity are retrieved. Finally, the selected image and catchphrase are combined to generate the meme.

2.2 Word subspace

The word subspace [7] was initially proposed to model textual information. In this representation, words are represented as word vectors in a real-valued feature vector space $\mathbb{R}^p$, by using the *word2vec*. With *word2vec*, it is possible to calculate the distance between two words, where words from similar contexts are represented by vectors close to each other, while words from different contexts are represented as different vectors [6]. To obtain the vector representation of words, we used a freely available *word2vec* model[1], trained by [6].

[1]https://code.google.com/archive/p/word2vec/

Therefore, given that words from a text belong to the same context, it is possible to compactly model the set of word vectors of each text as a word subspace.

Consider a text with N_t words. By translating each word into a vector using the *word2vec*, we obtain a set of word vectors $X_t = \{\boldsymbol{x}_k\}_{k=1}^{N_t}$. To model the word subspace from this set of word vectors, we first compute the following autocorrelation matrix, $\boldsymbol{R}_t$:

$$\boldsymbol{R}_t = \frac{1}{N_t} \sum_{i=1}^{N_t} \boldsymbol{x}_i \boldsymbol{x}_i^\top. \tag{1}$$

The orthonormal basis vectors of the m_t-dimensional subspace $\mathcal{Y}_t$ are obtained as the eigenvectors with the m_t largest eigenvalues of the matrix $\boldsymbol{R}_t$. The subspace $\mathcal{Y}_t$ is represented by the matrix $\boldsymbol{Y}_t \in \mathbb{R}^{p \times m_t}$, which has the corresponding orthonormal basis vectors as its column vectors.

2.3 Word Subspace from different medias

To model all three sources (images, catchphrases, and news text) into word subspaces in the same vector space, we first perform the following preprocessing:

Text data: For each input news text and catchphrase, we use the Stanford part-of-speech tagger[2] [9] to extract a set of meaningful words (verbs, nouns and adjectives). Then, only these words are translated to word vectors, using *word2vec*, resulting in sets of word vectors. The set of word vectors from an input text is denoted as $\{x_{txt}^i\}_{i=1}^{N_{txt}}$, while the vector set of a catchphrase is denoted as $\{x_{cphr}^i\}_{i=1}^{N_{cphr}}$.

Image data: Figure 2 shows our preprocessing for images. To make images compatible with text media, we represent them as sets of tags, which are extracted by using a deep neural network, the *AlexNet* [10]. Given a pre-defined set of tags, it extracts semantic information from the image in the form of a vector of probabilities among them. The N_{img} most likely tags are then converted to vectors using *word2vec*, primarily resulting in a set of tag vectors $\{x_{img}^i\}_{i=1}^{N_{img}}$ for each image. For tags with more than one word, e.g., "ice bear", we split it before translating to vectors, generating two separate tags, i.e., "ice" and "bear".

Under this setting, a set of words from a text contains syntactic information and can be seen as an ordered set; while the image tags are naturally non-ordered. By modeling both types of media as word subspaces, we can effectively represent the context of the corresponding media in the same vector space.

2.4 Retrieval based on word subspace

By representing all three types of media as word subspaces, we can compare them based on the similarity between the word subspaces. This way, it is possible to not only retrieve information from the same media

[2]https://nlp.stanford.edu/software/tagger.shtml (3.4.1)

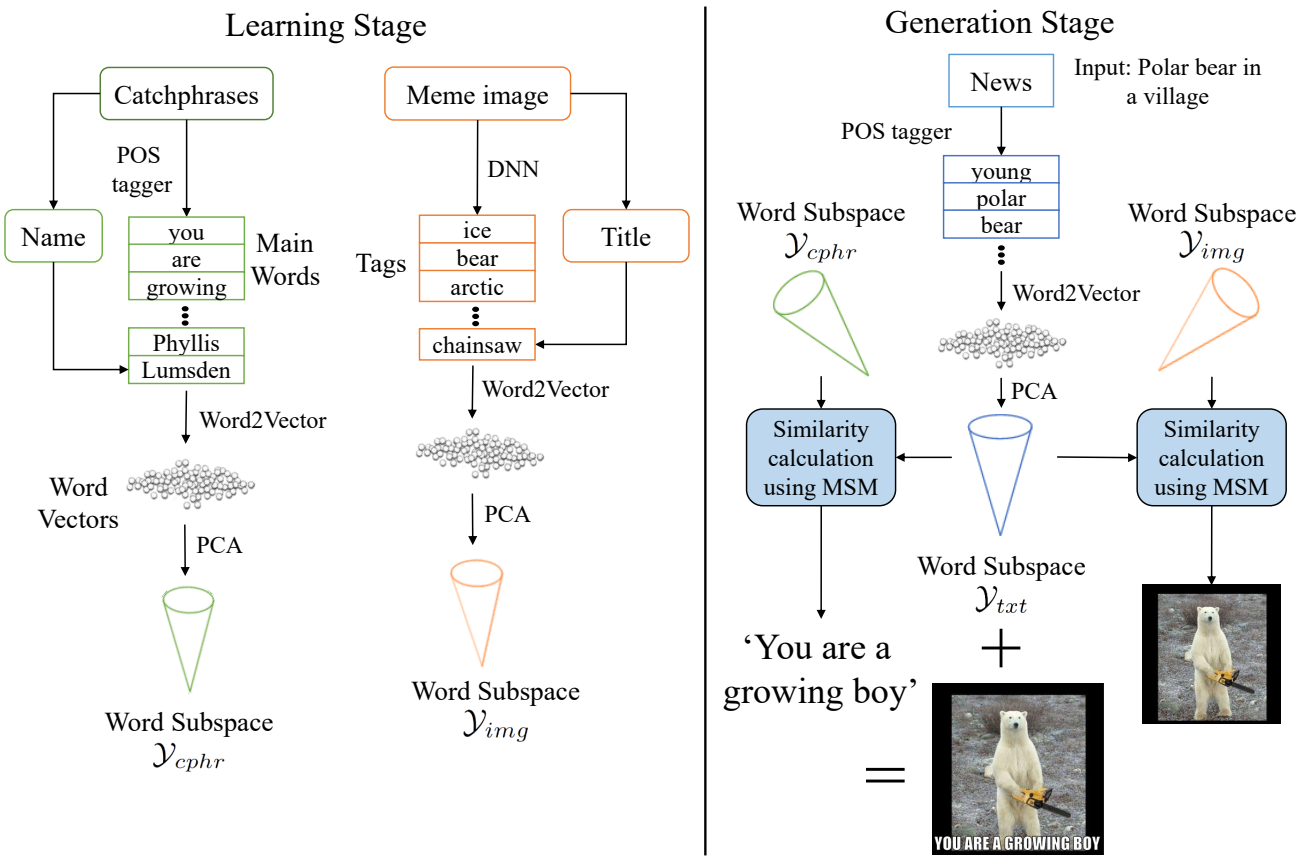


Figure 1. Flowchart of the proposed framework. Main words and image tags are extracted from catchphrases and news articles, using a POS tagger, and from meme images, using a DNN; Words and tags are then translated to vectors using the *word2vec* representation. Each set of word vector is modeled into a word subspace $\mathcal{Y}$, and the similarity between them is calculated using MSM.

format (e.g., a text from a text), but also retrieve information across formats (e.g., an image from a text).

Consider an input word subspace for news text data, $\mathcal{Y}_{txt}$, and a reference word subspace for image, $\mathcal{Y}_{img}$. We can compare them by measuring the canonical angles θ between them under the framework of MSM [8]. The canonical angles are defined as the arccosine of the singular values obtained by applying SVD [11] to the matrix $\boldsymbol{Y}_{txt}^{\top}\boldsymbol{Y}_{img}$, where $\boldsymbol{Y}_{txt} \in \mathbb{R}^{p\times m_{txt}}$ and $\boldsymbol{Y}_{img} \in \mathbb{R}^{p\times m_{img}}$ are the bases matrices of $\mathcal{Y}_{txt}$ and $\mathcal{Y}_{img}$, respectively. The similarity between these two word subspaces is defined by using t angles as follows:

$$S[t] = \frac{1}{t}\sum_{i=1}^{t}\cos^2\theta_i,\ 1 \le t \le m_{img}. \qquad (2)$$

Figure 3 shows the modeling and comparison of sets of words by MSM. This method can compare sets of different sizes, and naturally encodes proximity between sets that have common words or related words. For example, an input news text with the words "polar" and "bear" may be closer to an image word subspace containing the tag "artic" than an image word subspace containing the tags "bird" and "flower".

3 Experimental Evaluation

In this section, we discuss the validity of *news2meme* through two preliminary qualitative experiments. We first describe the databases we created in order to perform these experiments. Then, we describe the design of each experiment and summarize our main results.

3.1 Databases

We created three different databases:

- Meme image database S_{img}, with 812 images commonly used in memes, downloaded from the *Meme Generator* website [3]. For each image, we also extracted the image title.
- Catchphrase database S_{cphr}, with 1193 phrases taken from famous series, movies, and cartoons,

[3] https://imgflip.com/memegenerator

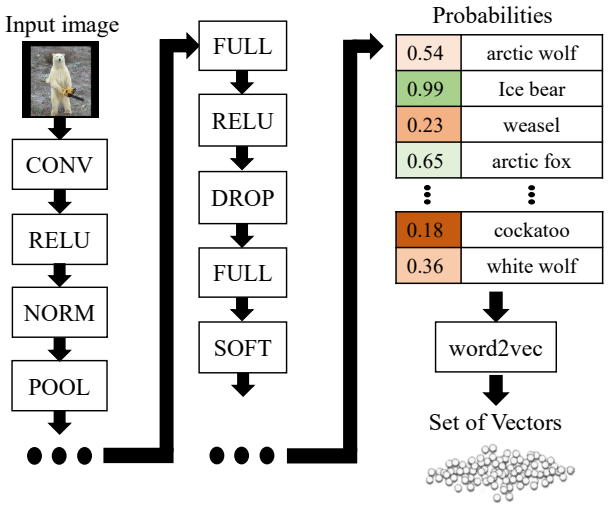


Figure 2. Flowchart of a deep neural network (DNN) for extracting a set of tag vectors from images. For tags with more than one word, we split it before translating to vectors, generating two separate tags.

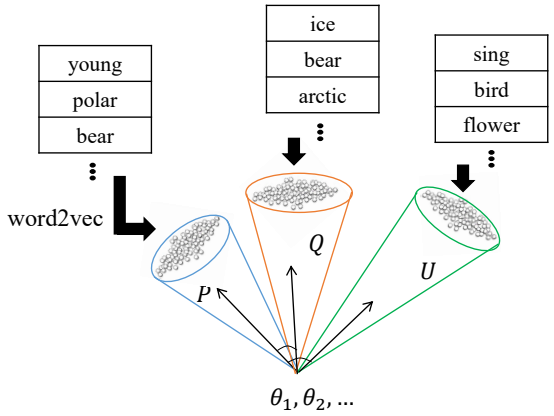


Figure 3. Comparison of sets of word vectors by the mutual subspace method.

downloaded from the *catchprase.info* website[4].

- News database S_{txt}, with 2517 news articles from the *News in Levels* website[5], from the categories 'History' (46 articles), 'Nature' (119), 'Sports' (93), 'Interesting facts' (1040), 'Funny' (61), and 'News' (1158). We chose this website over other news websites because it has short and simplified versions of news articles.

[4]http://www.catchphrases.info/
[5]http://newsinlevels.com

Table 1. Evaluation Results - General Score (%)

	General		
	Both Bad	One Good	Both Good
Sports	20.43	24.73	**54.84**
Nature	17.65	31.93	**50.42**
History	41.30	19.57	39.13
Interesting	29.45	44.36	26.18
News	71.21	16.67	12.12
Overall	30.66	37.18	32.15

3.2 Meme Generation

2517 memes were generated from news articles in the news database S_{txt} using our proposed framework. As training data, we used image tags extracted from images in the database S_{img} and main words from catchphrases in the database S_{phr}.

For the image tags, we considered the top 5 predictions by *AlexNet* for each image, and added words from the image title. As for the catchphrases, we extracted the main words (nouns, adjectives, and verbs) using the POS tagger and added the name of the character who says the catchphrase.

These sets of words and tags were then translated to vectors, ensuring that only one occurrence of each word was kept. PCA was then applied to each set of vectors, thus creating 812 meme image word subspaces $\{\mathcal{Y}^i_{img}\}^{812}_{i=1}$ and 1193 catchphrase word subspaces $\{\mathcal{Y}^i_{cphr}\}^{1193}_{i=1}$. We set the dimensions of the word subspaces, $\mathcal{Y}_{img}$ and $\mathcal{Y}_{cphr}$, to values ranging of from 4 to 7 and from 3 to 8, respectively.

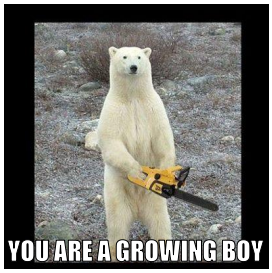
We used the news in S_{txt} as inputs, generating one meme for each news as described in Section 2. The word subspaces $\mathcal{Y}_{txt}$ dimensions ranged from 7 to 38.

3.3 Qualitative Experiments and Results

This preliminary experiment consists of two different parts: a meme evaluation experiment, designed to assess how well they matched the news, and a representativeness experiment, designed to verify whether our memes would be preferred over a random generation.

Meme evaluation experiment: We used 990 memes generated following the procedure in Section 3.2. Nine subjects were asked to read 110 news articles each and evaluate their corresponding generated memes with regards to the image and phrase. Subjects voted them as 'Good' when they depicted well the news and 'Bad' when they did not. To better understand the image and phrase combination, we gave a score for each meme. Considering 'Bad' as 0 and 'Good' as 1, we summed both votes, obtaining a maximum score of 2 when both image and phrase were 'Good' and a minimum score of 0 when both were 'Bad'. Table 1 shows the percentage of the scores received by each category.

Memes with the highest scores were from the 'Sports' and 'Nature' categories. We can see an example in Fig. 4(a), generated from a news about a



(a) 'Polar Bear in a Village'.

(b) 'Spider inside a man's body'.

Figure 4. Example of: (a) Good Meme (b) Bad meme.

young polar bear that strayed into a village in the Russian Arctic[6]. *News2meme* successfully related the news with a polar bear picture[7] and found a connection between "young" and "boy". In contrast, memes from the 'News' category had the lowest scores. Most of these news reported tragic events, which were perceived as not suitable for memes by the subjects.

Participants also reported cases where the generated meme had a 'Good' image, but a 'Bad' catchphrase and vice-versa. One example can be seen in Fig. 4(b), which was generated from an article about a spider inside a man's body[8]. The phrase translates as what could be seen as the point of view of the spider. However, the picture shows a man falling in the water[9].

When analyzing in more depth why *news2meme* generated a certain meme for a certain news article, we could see that, for some of them, the relation among the news text, the retrieved image and catchphrase was straightforward. For example, the news entitled "Star Wars exhibition"[10] generated a meme with the image of *Yoda*, a movie character, and the catchphrase "Whose game is over?". This shows that *news2meme* related the character in the news (*"It includes an original Darth Vader suit, a* ***Yoda*** *puppet, [...]"*) with the image. The catchphrase was related based on the word "game", which also appears in the news article (*"Visitors to the exhibition can also play a* ***game****, [...]"*).

On the other hand, some of the generated memes at first did not seem to have any relation to their news. However, when looking at the word vectors, we saw an indirect relation. For example, when inputting the news article entitled "Horse vs. Human", *news2meme* generated a meme with an image of Goku, an animation character, with the catchphrase "He is right behind me, isn't he?". This article is about a professional sprinter that ran faster than a racehorse[11]. *News2meme* related it to the character *Goku*, from the Japanese animation *Dragon Ball*. Among the most similar words to "Goku", we can find "dragon", due to animation's name. If we go further and look at the similar words for "dragon", we can find "fast", which relates to the news text (*"He is one of the* ***fastest*** *European sprinters, but can he beat a racehorse?"*). Therefore, although "Goku" and "fast" were not directly connected, the subspace representation encoded their indirect relation.

We also observed how the tags extracted by the DNN helped producing meaningful memes. In general, the extracted tags were essential for creating the appropriate meme. For example, when inputting a news about a nest of birds in a Christmas tree, entitled "Birds in a Christmas tree"[12], *news2meme* generated a meme with an image of a hawk with the catchphrase "This could be my big break". If *news2meme* had used only the meme image title (i.e., "hawkward"), it would not be able to associate this image to the news. However, the DNN extracted the tags "quinch", "quail", and "robin" that refer to types of birds and relate to the news article (*"A mother* ***bird*** *built a nest [...]"*).

However, there were cases where the tag extraction was inaccurate, resulting in a not meaningful meme. For instance, for the news article entitled "Animals in tourism"[13], *news2meme* generated a meme with an image of a strong man with catchphrase "You can do it, we can help". The DNN extracted tags such as "elephant" and "hippopotamus", which were related to the main topic of the article: animals (*"****Animal*** *attractions are a part of many holidays [...]"*). However, the image did not contain any of the tag elements and, thus, was not related to the article. This shows that, despite the improvement in many of the generated memes, in some cases using a pre-trained DNN can degrade the quality of the generations as it might just insert noise to the process.

Representativeness experiment:

In this experiment, our main goal was to determine whether memes generated by our framework could represent news articles content better than randomly generated memes. We designed a questionnaire where participants were asked to read ten different news articles and choose among four different options which meme better represented them. One of the memes was generated by our framework (generated memes), and the other three were randomly generated (random memes). We randomly chose these articles from 'Sports', 'Nature', 'History', 'Funny' and 'Interesting Facts' cate-

[6]https://www.newsinlevels.com/products/polar-bear-in-a-village-level-2/

[7]https://imgflip.com/memegenerator/Chainsaw-Bear

[8]https://www.newsinlevels.com/products/spider-inside-a-mans-body-level-2/

[9]https://imgflip.com/memegenerator/Nailed-It

[10]https://www.newsinlevels.com/products/star-wars-exhibition-level-2/

[11]https://www.newsinlevels.com/products/horse-vs-human-level-2/

[12]https://www.newsinlevels.com/products/birds-in-a-christmas-tree-level-2/

[13]http://www.newsinlevels.com/products/animals-in-tourism-level-2/

Table 2. Representativeness experiment: Percentage of votes

With Generated Memes		With Created Memes	
Generated	Random	Created	Random
62.55	37.45	76.86	23.14

gories of our news database S_{txt}.

Because the memes were generated based on a news article input, it was expected that participants would prefer them over the random ones. However, there was the possibility of participants showing no preference. This could be due to a flaw in our framework, or because the participants were choosing randomly. Therefore, we also showed ten news articles with four options, one of which was created by humans from the article (created memes), while the other three were random ones. Participants were not aware of the presence of the created memes and articles with generated and created memes were shown in random order.

This questionnaire was implemented as an online form, totaling 51 evaluations. Table 2 shows these results. Created memes were preferred by 76.86% of the participants, which indicates that they were not making random choices. 62.55% of the participants preferred the generated memes over the random ones. While it is clear that created memes are superior, this result shows that our memes are more meaningful than randomly generated ones.

4 Conclusions and Future Directions

In this paper, we proposed the *news2mem*, a framework for generating macro image memes from news articles. To solve this problem, we compared and matched three different media formats: a meme image, a catchphrase, and a news text. Our key idea is to extract tags from images, using a DNN, and main words from texts, using a POS tagger. Then, we represent these sets of tags and words as word subspaces. To compare them and retrieve the most suitable image and catchphrase to a news text, we used the MSM.

Our experiments showed that news articles containing tragic stories were perceived as not suitable for memes. When using news articles not related to tragic events, generated memes were preferred over random generations. This result shows that our framework can handle news articles and unconstrained images.

As future work, we seek improvements in two different levels: In the application level, we wish to apply a supervised technique where feedback information can be employed to enhance meme generation. In the algorithm level, we wish to comprehend better how to analyze word vectors in terms of sequence and further evaluate our framework in a multimedia retrieval context. Finally, we plan to perform experiments with different metrics (e.g., Likert scale) and more participants to better evaluate the generated memes.

Acknowledgment

This work is supported by the Japanese Ministry of Education, Culture, Sports, Science, and Technology (MEXT) scholarship.

References

[1] M. Knobel and C. Lankshear, "Online memes, affinities, and cultural production," in *A new literacies sampler*, vol. 29, pp. 199–227, 2007.

[2] P. Davison, "The language of internet memes," in *The social media reader*, pp. 120–134, JSTOR, 2012.

[3] D. Costa, H. G. Oliveira, and A. M. Pinto, "In reality there are as many religions as there are papers-first steps towards the generation of internet memes.," in *Proc. of 6th International Conference on Computational Creativity, ICCC*, pp. 300–307, 2015.

[4] H. G. Oliveira, D. Costa, and A. Pinto, "One does not simply produce funny memes!–explorations on the automatic generation of internet humor," in *Proc. of 7th International Conference on Computational Creativity, ICCC*, pp. 238–245, 2016.

[5] W. Y. Wang and M. Wen, "I can has cheezburger? a nonparanormal approach to combining textual and visual information for predicting and generating popular meme descriptions," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 355–365, 2015.

[6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[7] E. K. Shimomoto, L. S. Souza, B. B. Gatto, and K. Fukui, "Text classification based on word subspace with term-frequency," in *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2018.

[8] K. Fukui and A. Maki, "Difference subspace and its generalization for subspace-based methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 11, pp. 2164–2177, 2015.

[9] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proc. of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pp. 173–180, Association for Computational Linguistics, 2003.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[11] K. Fukui and O. Yamaguchi, "Face recognition using multi-viewpoint patterns for robot vision," *Robotics Research, The Eleventh International Symposium, ISRR*, pp. 192–201, 2005.