IEICE TRANS. INF. & SYST., VOL.E106–D, NO.5 MAY 2023

PAPER Special Section on Data Engineering and Information Management

Effective Language Representations for Danmaku Comment Classification in Nicovideo

Hiroyoshi NAGAO[†a)], **Koshiro TAMURA**[†b)], ***Nonmembers***, ***and*** **Marie KATSURAI**[†c)], ***Member***

SUMMARY Danmaku commenting has become popular for co-viewing on video-sharing platforms, such as Nicovideo. However, many irrelevant comments usually contaminate the quality of the information provided by videos. Such an information pollutant problem can be solved by a comment classifier trained with an abstention option, which detects comments whose video categories are unclear. To improve the performance of this classification task, this paper presents Nicovideo-specific language representations. Specifically, we used sentences from Nicopedia, a Japanese online encyclopedia of entities that possibly appear in Nicovideo contents, to pre-train a bidirectional encoder representations from Transformers (BERT) model. The resulting model named Nicopedia BERT is then fine-tuned such that it could determine whether a given comment falls into any of predefined categories. The experiments conducted on Nicovideo comment data demonstrated the effectiveness of Nicopedia BERT compared with existing BERT models pre-trained using Wikipedia or tweets. We also evaluated the performance of each model in an additional sentiment classification task, and the obtained results implied the applicability of Nicopedia BERT as a feature extractor of other social media text.

key words: *comment classification, Danmaku, BERT, Nicovideo*

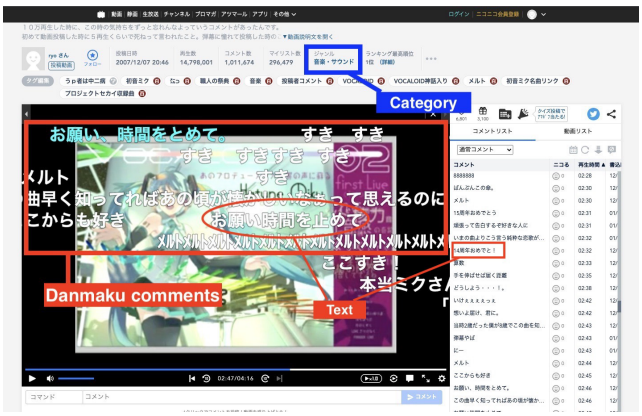


Fig. 1 Example of Danmaku comments on Nicovideo.

1. Introduction

Social media services, which are expanding their influence along with the increase in users, have greatly changed the way of offering information. Video-sharing platforms, such as YouTube* and Nicovideo** (or Nico Nico Douga), enable content sharing among large numbers of users through video posting and live streaming. On these sites, users post various opinions and impressions of videos through the commenting function, facilitating communication between contributors and viewers. Additionally, viewers can freely enjoy conversation with each other through the exchange of comments. There are various ways to display comments. For example, services such as Nicovideo and Bilibili*** have a real-time commenting function as shown in Fig. 1. The state in which comments fill the video playback screen or the comments themselves are called "Danmaku," derived from a Japanese word used in shooting games. In this paper, we term each comment that moves on the screen a Danmaku comment.

Although Danmaku commenting promotes communication among users and provides an entertaining experience of co-viewing, it also has a problem of *excess of information* and *information pollutants*: there might be too many comments, including those irrelevant to the content, such as posts about personal matters and disputes between users. These can contaminate the quality of video information, resulting in the cognitive load especially for infrequent users. Thus, it is crucial to develop a comment filtering mechanism that a user can use as they like [1]. Conventional research on video comment filtering has often determined the sentiment polarity of a comment sentence and eliminated negative comments [2]. While this approach can reduce discriminatory and slanderous comments, it still cannot filter out comments irrelevant to the video's content. Thus, we tackled the challenge of classifying Danmaku comments into video categories. For example, given a video labeled with "sports," extracting sports-related comments from the video can enrich the viewing experience and contribute to data mining systems.

Due to the Danmaku characteristics described earlier, not all video comments can be clearly categorized into specific categories. In our preliminary study [3], we proposed a framework that abstained from judging comments that did not fall into any predefined class or that were difficult to narrow down to a single class. The experiments conducted using Nicovideo comments showed the effectiveness of such classification. In [3], the comment classifier used

Manuscript received June 27, 2022.
Manuscript revised November 16, 2022.
Manuscript publicized January 16, 2023.

[†]The authors are with the Doshisha University, Kyotanabe-shi, 610–0394 Japan.
a) E-mail: nagao21@mm.doshisha.ac.jp
b) E-mail: koshiro.tamura@mm.doshisha.ac.jp
c) E-mail: katsurai@mm.doshisha.ac.jp

DOI: 10.1587/transinf.2022DAP0010

*https://www.youtube.com/
**https://www.nicovideo.jp/
***https://www.bilibili.com/

language representations learned from Japanese Wikipedia[†] to calculate comments' textual features. However, the text in Wikipedia requires a formal tone and does not cover words, phrases, and sentences, which the Nicovideo user community prefers. To solve this domain gap problem, we investigated the effectiveness of language representations obtained from a large-scale text resource in the domain compatible with Nicovideo. Specifically, we used Nicopedia[††], a Japanese online encyclopedia of entities that possibly appear in Nicovideo, as a sentence corpus to pre-train a bidirectional encoder representations from Transformers (BERT) model [4]. We term the resulting encoder "Nicopedia BERT" in this paper and use it as a feature extractor in the video comment classification task. To evaluate the performance of Nicopedia BERT, we conducted experiments on a real video comment dataset and an external sentiment analysis dataset. The results demonstrated that Nicopedia BERT achieved the best performance in video comment classification. We also found that it showed moderate results even in a tweet sentiment analysis task, implying its usefulness in obtaining comment embeddings in a casual tone.

The main differences between this paper and the previous report [3] are twofold:

- We present a novel BERT model trained using Nicopedia with the details of its text preprocessing. The codes are available on the Web.[†††]
- The present paper contains additional experiments of visualization, comparison with BERT models trained using Japanese Wikipedia or tweets, and an application to sentiment analysis on social media; these demonstrate the effectiveness of Nicopedia BERT.

The remainder of this paper is structured as follows: Sect. 2 introduces previous work on language representations based on large-scale text data and video comment classification. Section 3 details the datasets used in this study and their preprocessing. Section 4 describes a method of learning language representations and our comment classification framework. Section 5 presents the usefulness of the proposed method through experiments on comment classification, visualization of language representations, and tweet sentiment polarity classification. Finally, Sect. 6 provides conclusions and suggests possible directions for future work.

2. Related Work

Natural language processing techniques have been widely used to develop various applications in social media services. Representing the meanings of words in a sentence as a vector of numerical values is crucial for exploiting machine learning methods. A classical method is to count word co-occurrence frequencies in a text document based on the distributional hypothesis, which has a high computational cost when using a large-scale corpus. In contrast, neural network-based methods, such as word2vec [5], can compute efficiently even for large corpora by sequential inference processing and can learn the contextual information. Such neural network-based methods have recently become mainstream, especially since the announcement of Transformer [6]. A typical method is BERT, which pre-trains a model for contextual embeddings of sentences using a large corpus and then fine-tunes the model on a relatively small amount of data in a target task. It has achieved significant performance improvement over conventional language representations in various tasks related to text data. Thus, our study uses BERT as a base architecture of sentence embeddings to recognize Danmaku comments in Nicovideo.

There exist diverse models pre-trained in specific domains. For example, large corpora of papers published in computer science and biomedicine have been used to train scientific BERT models [7], [8]. Their language representations improved the performance of academic tasks, such as named entity recognition and relation extraction. Sung et al. [9] used textbooks to improve the BERT pre-training step for the subsequent task of automatic short answer grading. Dai et al. [10] developed TweetBERT and ForumBERT, which were pre-trained using tweets and forum text on business review, respectively, and showed the benefits of these two resources. These conventional studies showed that domain-specific BERT models outperformed a standard model trained using a generic corpus. Motivated by these prior works, we investigated whether Nicopedia could empower the language representations for Danmaku comment categorization in Nicovideo.

There are also some examples of training BERT using Japanese texts. Shibata et al. [11] pre-trained BERT using Japanese Wikipedia and presented its application to parsing. Other research groups also released BERT models pre-trained using Japanese Wikipedia [12], [13]. Sakaki et al. [14] constructed a BERT model using Japanese tweets collected from Twitter[††††] and used it in tweet sentiment analysis. Kawazoe et al. [15] used Japanese clinical text stored in an electronic health record system to pre-train BERT. The number of domain-specific BERT models for the Japanese language is still few, and our work is the first to use Nicopedia as a resource to obtain language representations.

Video comment filtering methods have been studied to maintain the quality of the information in video content or improve the user experience when watching videos. For example, Chen et al. [16] extracted TF-IDF features from YouTube comments and constructed a sentiment classifier. Bai et al. [17] used the model architecture of BERT to analyze the sentiment of Danmaku comments in videos. These methods were used to detect harmful content that should to be deleted; however, the sentiment-based approach is not always practical for removing irrelevant comments that make the content messy. Thus, we considered the problem of

[†]https://ja.wikipedia.org/
[††]https://dic.nicovideo.jp/
[†††]https://github.com/mm-doshisha/nicobert
[††††]https://twitter.com/

categorizing comment contents and abstained from judging comments that are difficult to categorize, thereby achieving a flexible filtering function.

3. Datasets and Preprocessing

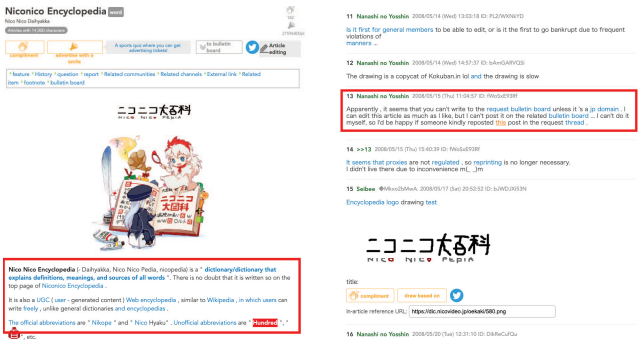
This section explains the details of Nicopedia data used for learning language representations and Nicovideo comment data used for video comment classification. We also provide the details of their preprocessing.

3.1 Nicopedia Dataset

The dataset for learning language representations is "Nicopedia data[†]," which contains all articles posted in Nicopedia from May 2008 to February 2014. Each article page in the dataset consists of a header, article body, and forum data. The header describes the article ID, title, kana, article type, creation date, etc. The article body data contains the corresponding article ID, text, and modification date; it is provided as an HTML source code. Forum data includes the corresponding article ID, comment number, posting date, and comment text. We extracted the text of the article body data and the forum data, whose example is shown in Fig. 2, and applied the following three preprocessing modules:

- Common preprocessing:
  - Remove URLs.
  - Apply NFKC normalization.
  - Remove white space at the beginning and end of sentences.
  - If the same character appears three or more times in a single comment, replace it with two characters (e.g. wwwww → ww).
- Preprocessing of article body:
  - Remove HTML tags.
  - Remove sections that do not directly describe the entity, such as footnotes and links to related articles.
- Preprocessing of forum:
  - Remove the mentions such as ">>" at the beginning of a sentence and the number of the destination of the mentions.
  - Remove sentences with less than 10 characters after preprocessing.

After preprocessing, We divided the text into sentences by punctuation marks or line feeds. The resulting corpus comprised 3.3 million sentences from the text of articles and 10 million sentences from the forum items, totaling approximately 13 million. It contained many commentary sentences with colloquial expressions and slang words. The articles' text also included many terms that are frequently used on the Internet. Japanese Wikipedia, a representative example of a large text corpus, uses polite expressions and has no user forum. Therefore, our Nicopedia can be a suitable training resource for language representations of Nicovideo applications compared with Japanese Wikipedia.



(a) Article body. (b) User forum item.

Fig. 2 Sentence sources of a Nicopedia article page, in which original Japanese sentences were translated into English.

3.2 Nicovideo Dataset

Our classifier training resource is "Nicopedia comment data[††]", which consists of metadata and comments of videos posted on Nicovideo from March 2007 to November 2018. The video metadata contains the title, tags, and posting date, whereas the comment data contains the comment sentences and posting date. In this study, we extracted from this dataset a total of 60,000 comments; 10,000 each for videos in the six categories of "politics," "cooking," "animals," "trains," "sports," and "games," posted in the year to November 2018.

In Nicovideo, each comment is usually written in Japanese but could contain English words. We applied the following preprocessing to the comments.

- Replace English words with lowercase letters.
- Replace full-width alphabets with half-width ones.
- Remove symbols and pictograms.
- Remove URLs.
- Remove sentences with less than eight characters.
- Replace all digits with 0.
- If the same character appears three or more times in a row, replace it with two characters.

Ideally, for classifier training, each comment should be accurately labeled with its topic category. However, it is usually extremely time-consuming to prepare the ground truth of a large number of comments. Furthermore, since there is no guarantee that successive comments are similar to each other, labeling comment pairs with positive or negative is also difficult. Thus, we defined the topic category of a training video as the pseudo label for all Danmaku comments posted on that video. For example, if a video's topic category is sports, the pseudo label for all comments appearing in the video is considered sports. The role of these

[†]https://www.nii.ac.jp/dsc/idr/nico/nicopedia-apply.html

[††]https://www.nii.ac.jp/dsc/idr/nico/nicocomm-apply.html

pseudo labels is explained in Sect. 4.2.

4. Methodology

This section describes the construction of Nicopedia BERT and its application to Danmaku comment classification in Nicovideo. Figure 3 shows a schematic diagram of the proposed framework. The pre-training and comment classification steps are based on the datasets in Sect. 3.1 and Sect. 3.2, respectively.

4.1 Pre-Training BERT Based on Nicopedia

BERT requires converting sentences into a sequence of word indexes. Since Japanese sentences do not have separators like spaces in English, we need to divide the sentences into words via morphological analysis. MeCab [18] is a well-known Japanese morphological analysis tool based on a predefined dictionary, but it suffers from unknown words. Recently, a subword-based method named SentencePiece [19] has been proposed. It divides low-frequency words into smaller units (called subwords) directly from the raw text without morphological analysis. This approach effectively reduces the number of unknown words. Thus, we used SentencePiece as a text tokenizer.

There are two tasks in BERT pre-training: Masked Language Model (Masked LM) and Next Sentence Prediction (NSP). Masked LM first randomly selects 15% of the words in a sentence and replaces them with a special token [MASK]. It then predicts what the original token was before it was masked. For two consecutive sentences, A and B, NSP replaces B with a random sentence in the corpus with a probability of 50% and predicts whether A and B are consecutive or not. NSP assumes semantic relationships between original subsequent sentences. However, considering the forum items in our Nicopedia dataset, although each comment sentence is often relevant to the corresponding article's entity, it does not always have relatedness with its previous and subsequent comments. Recent studies also reported that removing the NSP task improved the performance of downstream tasks [20]. Therefore, in this study, we used only Masked LM as the pre-training task. The resulting pre-trained model is referred to as Nicopedia BERT.

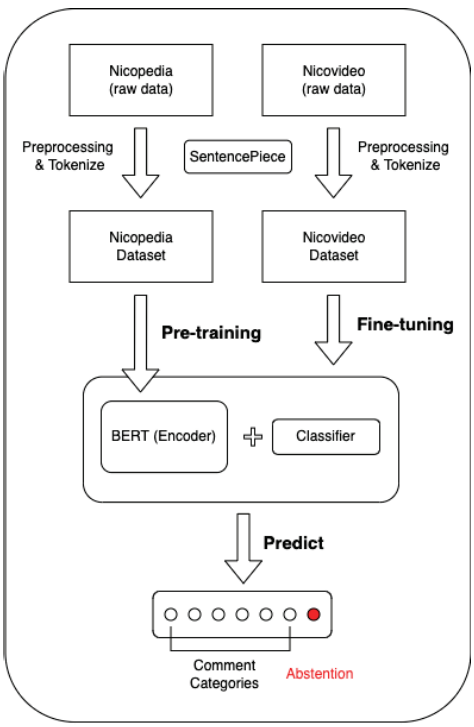


Fig. 3 Overview of the proposed framework.

4.2 Application to Danmaku Comment Classification

This subsection describes two approaches of Danmaku comment classification with an abstention option [3]: *Softmax Response* and *Deep Gamblers*. In both approaches, we added a fully-connected layer as a category predictor to the pre-trained BERT model: it converts the 768-dimensional BERT embeddings to vectors of prediction results. We then fine-tuned the weights of the entire model using each method's particular loss function. During the training phase, we let the model learn discriminative features using pairs of comment texts and their pseudo labels assigned in Sect. 3.2.

Softmax Response: The Softmax Response (SR) [21] approach used the cross-entropy loss to train a classifier. Considering the maximum Softmax value of the model output as the abstention confidence, it abstained from the decision if the value was lower than a predefined threshold.

Deep Gamblers: Instead of cross-entropy loss, we introduced a loss called deep gamblers (DG) [22] to consider the confidence of the prediction. Specifically, when the number of categories of videos is m ($m = 6$ in our dataset), a new class called "abstention" is added to the m-class classification problem, formulated as an $m + 1$ class classification problem using the following loss function:

$$\max_{\boldsymbol{w}} \sum_{x \in \mathrm{B}} \log[f_{\boldsymbol{w}}(x)_{j(x)} o + f_{\boldsymbol{w}}(x)_{m+1}], \tag{1}$$

where f indicates the BERT model with category classifier, x is an input text, $\boldsymbol{w}$ denotes the parameters of f, $j(x)$ is the class label of x, B is a mini-batch of training samples, and o is a hyperparameter that represents the degree of abstention. The o can be set in the range $1 < o < m$. It controls the incentive for the model to abstain from prediction, with higher o encouraging abstention. We define $f_{\boldsymbol{w}}(x)_c$ as the output of the category classifier applied with the softmax function, which represents the predicted probability for class c. The $m + 1$-th class means the abstention class, so the $f_{\boldsymbol{w}}(x)_{m+1}$ represents the rejection score for each input data. In the DG approach, the classifier can be trained to increase the rejection score of data that is judged to not be properly labeled during training. This approach allows for bolder labeling, which reduces the effort required for annotation.

Our comment filtering approach introduces the concept

of *coverage*, which is the ratio of the testing data classified into specific categories to all testing data. The coverage controls how many predictions we abstain from. For example, suppose the coverage is 0.7 in a classification experiment on testing data. In that case, we classify only the top 70% of data with the lowest abstention confidence into predefined categories and abstain from judging the remaining 30% of data. Irrelevant comments may be left behind with high coverage, while With low coverage, only comments that can be confidently regarded as relevant will remain. In practical use, users can adjust the coverage level to what extent they will have a good viewing experience.

Training a comment-category classifier contributes to discriminating specific comments appearing in a particular category from comments that are common between several categories. This labeling is useful for not only filtering comments but also mining informative scenes in a target category.

5. Experiments

This section presents the results of the experiments to verify the effectiveness of the language representations obtained from the Nicopedia dataset. For pre-training BERT, we used HuggingFace implementation[†]. We set the vocabulary size of SentencePiece to 32,000. We used eight GPUs per node (the total memory of 360 GB), and trained on four nodes in parallel. The number of training steps was 1,200,000, corresponding to 23 epochs, and took about four days.

To examine the performance of different training corpora for building a BERT model, we compared the following two existing models: (i) **Wikipedia BERT** [13] trained on Japanese Wikipedia and (ii) **Twitter BERT** [14] trained using 85 million Japanese tweets.

5.1 Results of Danmaku Comment Classification

Since the comments in our dataset had just pseudo labels, to quantitatively evaluate the performance of Danmaku comment classification, we manually prepared the ground truth of category labels as follows: we first split the 60,000 video comments in the dataset into 48,000 training data and 12,000 testing-comment candidates. From the candidate data, we randomly extracted 350 comments belonging to each video category. Then, two undergraduate students (who were not involved in this study and had many experiences in video-sharing websites) manually assigned each comment with a class label. The two annotators estimated which of the six video categories each comment was posted on, and if it was difficult to determine, they could choose to abstain. The label agreement rate and the kappa coefficient between the two annotators for the 2,100 comments was 0.693 and 0.627, respectively. We used comments whose labels were matched between the two annotators as a ground truth dataset. Table 1 shows the number of testing comments used for each category in the constructed evaluation dataset.

[†]https://github.com/huggingface/transformers

Table 1 The number of comments for each category for which the judgments of the two annotators matched.

Category	Number of comments
Politics	198
Cooking	213
Animal	101
Train	149
Sport	193
Game	203
Abstention	398

In comment classifier training, we used Adam [23] as an optimizer, a batch size of 32, and an epoch of 12. To stabilize the learning, we used cross-entropy loss for the first three epochs and gambler loss for the remaining nine epochs, following the work of Liu et al. [22]. Here, we extracted 2.2% of the training data as the validation data, in which we searched for the value of hyperparameter o of DG as well as the threshold of SR for each BERT model. The selection process for the optimal parameters was as follows: we first found the best value for each of the four metrics (i.e., accuracy, recall, precision, F1). Then, as an approach consistent among DG and SR, taking a majority vote of these evaluation metrics determined the final parameter value. We tested the values of o from 1.0 to 6.0 with a step size of 0.5. As a result, we selected 4.5, 5.0, and 5.5 for Wikipedia, Twitter, and Nicopedia, respectively. We also searched for the threshold value of SR between [0.1, 0.9] with a step size of 0.1 and chose a value of 0.2 for all models.

As evaluation metrics for classification performance, we calculated accuracy, recall, precision, and F1 for different coverages. We used macro average for the precision, recall and F1 following the experimental settings in [3]. Table 2 shows the results of the comparison between the conventional models and the proposed model for each evaluation metric. In the table, DG and SR indicate the method used for classification abstention, respectively. The result shows that Nicopedia BERT obtained higher values for all metrics and coverages compared with the other two BERT models. Interestingly, although tweets seem relevant to languages used in Nicovideo, there was a large difference in performance between Twitter BERT and Nicopedia BERT. These results clarified the importance of training language models on sentences from the same domain with Nicovideo for Danmaku comment recognition. For Nicopedia BERT and Twitter BERT, in common with all the metrics, DG wins at high coverage while SR wins at low coverage. On the other hand, in Wikipedia BERT's results, SR performed slightly better than DG across all coverages. A possible practical case would be to switch DG and SR flexibly; for example, one can use DG for high coverages (0.9 to 1.0) and SR for low coverages. Through the results, we observed that the lower the coverage, the higher the evaluation value. However, again, it should be determined by the user's qualitative evaluation as to how much the coverage should be reduced.

We also investigated the effects of out-of-vocabulary

Table 2 Evaluation results of comment classification for each metric.

Accuracy							Precision					
Wikipedia		Twitter		Nicopedia		Coverage	Wikipedia		Twitter		Nicopedia	
DG	SR	DG	SR	DG	SR		DG	SR	DG	SR	DG	SR
0.7162	0.7228	0.7001	0.6868	**0.8269**	0.8070	1.00	0.7052	0.7125	0.6908	0.5783	**0.8176**	0.7965
0.7363	0.7463	0.7214	0.7164	**0.8468**	0.8299	0.95	0.7240	0.7350	0.7088	0.7047	**0.8382**	0.8183
0.7563	0.7626	0.7426	0.7342	**0.8582**	0.8529	0.90	0.7429	0.7491	0.7289	0.7230	**0.8486**	0.8412
0.7720	0.7887	0.7553	0.7542	0.8710	**0.8776**	0.85	0.7564	0.7766	0.7387	0.7417	0.8601	**0.8679**
0.7920	0.8038	0.7695	0.7813	0.8806	**0.8948**	0.80	0.7746	0.7905	0.7500	0.7671	0.8662	**0.8866**
0.8134	0.8235	0.7818	0.8071	0.8827	**0.9117**	0.75	0.7950	0.8104	0.7622	0.7913	0.8654	**0.9014**
0.8419	0.8432	0.8014	0.8284	0.9014	**0.9284**	0.70	0.8232	0.8318	0.7827	0.8124	0.8849	**0.9207**

Recall							F1					
Wikipedia		Twitter		Nicopedia		Coverage	Wikipedia		Twitter		Nicopedia	
DG	SR	DG	SR	DG	SR		DG	SR	DG	SR	DG	SR
0.7237	0.7278	0.7030	0.5901	**0.8310**	0.8107	1.00	0.7096	0.7147	0.6909	0.5791	**0.8202**	0.7985
0.7412	0.7477	0.7226	0.7186	**0.8488**	0.8321	0.95	0.7288	0.7362	0.7113	0.7041	**0.8403**	0.8205
0.7576	0.7616	0.7435	0.7335	**0.8572**	0.8503	0.90	0.7466	0.7506	0.7316	0.7200	**0.8500**	0.8426
0.7710	0.7859	0.7519	0.7517	0.8689	**0.8727**	0.85	0.7602	0.7770	0.7407	0.7398	0.8622	**0.8674**
0.7856	0.7963	0.7622	0.7767	0.8729	**0.8886**	0.80	0.7769	0.7879	0.7524	0.7651	0.8678	**0.8854**
0.8018	0.8107	0.7724	0.7987	0.8715	**0.9051**	0.75	0.7949	0.8045	0.7637	0.7896	0.8665	**0.9008**
0.8247	0.8265	0.7898	0.8209	0.8893	**0.9217**	0.70	0.8213	0.8219	0.7833	0.8114	0.8856	**0.9192**

Table 3 OOV rate of tokens for each datasets.

Model	Train			Test		
	OOV Tokens	Total Tokens	Token Rate	OOV Tokens	Total Tokens	Token Rate
Wikipedia	512	600,012	0.09%	15	14,414	0.10%
Twitter	457	380,094	0.12%	15	9,681	0.15%
Nicopedia	115	540,185	0.02%	0	13,273	0.00%

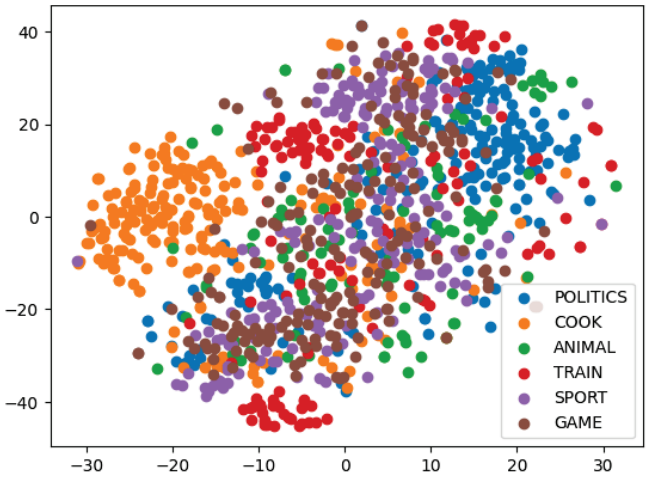
Table 4 OOV rate of sentences for each datasets.

Model	Train			Test		
	OOV Sentences	Total Sentences	Sentence Rate	OOV Sentences	Total Sentences	Sentence Rate
Wikipedia	390	47,315	0.82%	15	1,057	1.42%
Twitter	406	47,315	0.86%	15	1,057	1.42%
Nicopedia	82	47,315	0.17%	0	1,057	0.00%

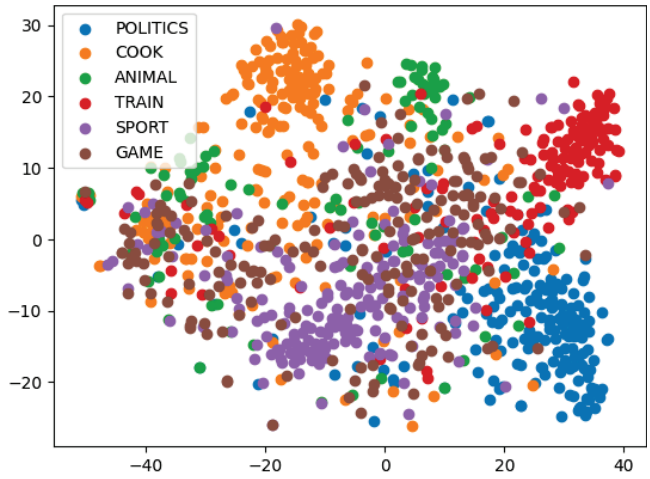
(OOV) using an experiment similar to that conducted in [24]. Specifically, we counted the number of OOV tokens and the number of sentences in which an OOV token appeared at least once in the sentence. Table 3 and Table 4 show the number and the rate of OOV tokens and OOV sentences, respectively. Experimental results show that the Nicopedia model (tokenizer) has the lowest OOV rate for both tokens and sentences. This was especially noticeable in the testing data, where the numbers of OOV tokens and OOV sentences in Nicopedia model were zero. Thus, our BERT is more suitable for Danmaku comment classification than the conventional models.

5.2 Visualization of Language Representations

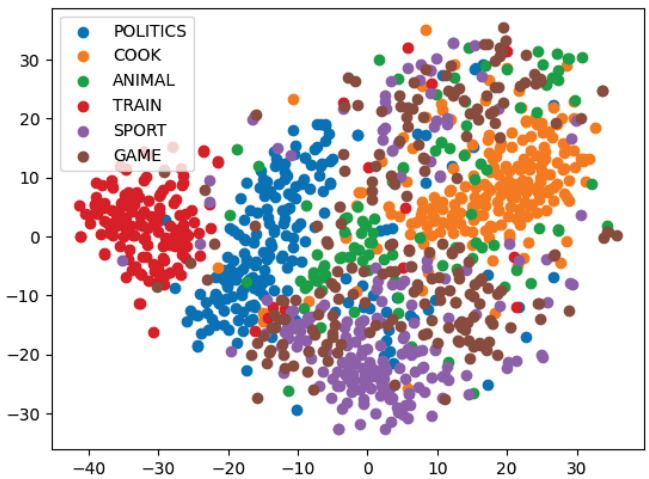
We further investigate how each BERT model discriminates Danmaku comment categories using the *t*-SNE algorithm [25]. Figure 4 shows the two-dimensional visualization results of Nicovideo comment embeddings provided by different BERT models. In the figures, a point corresponds to a single comment sentence in the evaluation dataset, and the color of each point represents a corresponding video category. We found that embeddings from Nicopedia BERT formed clusters of video categories well. Focusing on the results of Wikipedia BERT, comments in different colors significantly overlap, except for "cooking" comments. Comparing the results between Twitter BERT and Nicopedia BERT, both models formed clusters of "politics" and "train" well, while Nicopedia BERT created a clear group of "cooking." Comments of "animal" and "game" were relatively scattered; the reason for it might be that these two categories have more comments that are somehow relevant to other video categories. Such visualization of the comment embeddings has the potential to facilitate the understanding of each video category's comment tendency; we will introduce the obtained insights into other Nicovideo-related tasks



(a) Result of Wikipedia BERT.



(b) Result of Twitter BERT.



(b) Result of Nicopedia BERT.

Fig. 4 Visualization results of each BERT model's comment embeddings.

in future work.

5.3 Evaluation of Model Applicability via Tweet Sentiment Polarity Classification

We conducted an additional experiment on tweet sentiment classification to evaluate the applicability of the obtained language representations. For the evaluation, we used the WRIME dataset [26], a Japanese tweet dataset annotated via crowdsourcing. The dataset includes five sentiment labels (i.e., strongly positive, positive, neutral, negative, and strongly negative) assigned from an objective viewpoint. In addition to the original five-class evaluation, we also constructed a three-class classification task, whose classes are "strong positive or positive," neutral, and "strong negative or negative." Table 5 shows the number of samples for each sentiment class, from which the numbers of samples in the three-class task can also be calculated. Out of 35,000 samples in the WRIME dataset, we used 2,500 samples divided in advance as testing data. Similarly to category classification in Sect. 4.2, we added a fully-connected layer to each pre-trained BERT encoder so that it worked as a classifier and fine-tuned the weights of the entire model. Following the experimental settings in [26], for classifier training, we used Adam and set the batch size to 32, the dropout rate to 0.1, and the learning rate to 2e-5. Early stopping was applied every three epochs up to 20 epochs.

Table 5 The number of sentiment polarity labels for each class.

Sentiment polarity	Number of data
strong positive	2245
positive	9138
neutral	11462
negative	10468
strong negative	1687
Total	35000

We calculated Accuracy, Precision, Recall, F1, mean absolute error (MAE), and quadratic weighted kappa (QWK) as evaluation measures. We used macro average for Precision, Recall and F1, following the experimental

Table 6 Results of five-class tweet sentiment classification.

Model	Accuracy	Precision	Recall	F1	MAE	QWK
Wikipedia BERT	0.553	0.523	0.470	0.487	0.502	0.700
Twitter BERT	**0.624**	**0.604**	**0.529**	**0.546**	**0.410**	**0.768**
Nicopedia BERT	0.614	0.603	0.486	0.488	0.420	0.754

Table 7 Results of three-class tweet sentiment classification.

Model	Accuracy	Precision	Recall	F1	MAE	QWK
Wikipedia BERT	0.700	0.687	0.690	0.688	0.338	0.702
Twitter BERT	0.748	0.742	0.744	0.742	0.267	**0.782**
Nicopedia BERT	**0.750**	**0.749**	**0.748**	**0.747**	**0.265**	0.781

settings in [26]. Table 6 and Table 7 show the performance comparisons of different BERT models on each of the evaluation metrics in five-class and three-class classification tasks, respectively. In Table 6, Twitter BERT showed the highest values for all evaluation metrics, and Nicopedia BERT outperformed Wikipedia BERT. It seems natural that Twitter BERT can obtain the best scores since its training resource's domain is the same as the target task's domain, as our comment classification experiments indicated. However, in Table 7, Nicopedia BERT achieved slightly better results for all the evaluation measures except for QWK than Twitter BERT. Wikipedia BERT showed the worst performance in both settings; this would be due to the lack of informal or colloquial phrases in Wikipedia. These results implied that Nicopedia could be a resource of text understanding in not only Nicovideo but also other social media data like tweets.

6. Conclusion and Future Work

This study presented Nicopedia-based language representations for classifying Danmaku comments in Nicovideo. The experiments conducted using actual Danmaku comment data showed that our Nicopedia BERT was the most suitable for the comment classification compared to the representations obtained from Japanese Wikipedia and tweets. Furthermore, the results of the additional tweet sentiment classification task implied Nicopedia BERT's applicability to represent other social media text with a casual tone.

Further room for investigation and improvement exists in our study. For example, we did not consider ordering relations (e.g., posting dates, mentions, or responses) of the forum items and decided not to use NSP. It would be interesting to introduce such metadata-based information of sentences to BERT training. The Nicopedia dataset used in this study can also be extended by adding recent articles. Although our pre-trained model has yielded compelling results, the computational cost of training based on million-scale data was very high. We will address this problem by incorporating improved models of the advanced BERT, such as ELECTRA [27], which can be efficiently trained even on highly large-scale corpora. Furthermore, we plan to develop a user-friendly filtering system using the classification method proposed in this study. Visualizing the comment embeddings with their video topics can also provide insights into each video category's tendency in comments, which will facilitate the development of applications in Nicovideo. The current comment classification approach can find category-specific comments and discard general comments that appear in diverse videos. The next challenge would be breaking the concept of comment filtering down into several aspects, not limiting to the relevance with video categories. Our BERT, which is the main focus of this paper, will be a base model in several filtering approaches.

Acknowledgements

This research was partly supported by JSPS KAKENHI Grant Number 20H04484. This study used "Nicopedia data" and "Nicovideo Comment etc. data", provided by DWANGO Co., Ltd. through the National Institute of Informatics Research Data Repository.

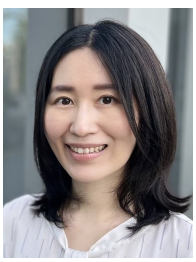
References

[1] Y. Chen, Q. Gao, and P.L.P. Rau, "Watching a movie alone yet together: Understanding reasons for watching danmaku videos," Int. J. Hum.-Comput. Interact., vol.33, no.9, pp.731–743, 2017.
[2] J. Savigny and A. Purwarianti, "Emotion classification on youtube comments using word embedding," 2017 International Conference on Advanced Informatics, Concepts, Theory, and Applications (ICAICTA), pp.1–5, 2017.
[3] K. Tamura and M. Katsurai, "Selective classification of Danmaku comments using distributed representations," The 23rd International Conference on Information Integration and Web Intelligence (iiWAS), pp.130–136, Nov. 2021.
[4] J. Devlin, M.W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional Transformers for language understanding," Proc. 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp.4171–4186, June 2019.
[5] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," International Conference on Learning Representations (ICLR), 2013.
[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," Advances in Neural Information Processing Systems, pp.5998–6008, 2017.
[7] I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A pretrained lan-

guage model for scientific text," Proc. 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp.3615–3620, Nov. 2019.
[8] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C.H. So, and J. Kang, "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," Bioinformatics, vol.36, no.4, pp.1234–1240, Feb. 2020.
[9] C. Sung, T. Dhamecha, S. Saha, T. Ma, V. Reddy, and R. Arora, "Pre-training BERT on domain resources for short answer grading," Proc. 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Nov. 2019.
[10] X. Dai, S. Karimi, B. Hachey, and C. Paris, "Cost-effective selection of pretraining data: A case study of pretraining BERT on social media," Findings of the Association for Computational Linguistics: EMNLP 2020, pp.1675–1681, Nov. 2020.
[11] T. Shibata, D. Kawahara, and S. Kurohashi, "Improved accuracy of Japanese parsing with BERT," Proc. 25th Annual Meeting of the Association for Natural Language Processing, pp.205–208, 2019.
[12] Tohoku NLP Group, "Pretrained Japanese BERT models," https://github.com/cl-tohoku/bert-japanese/tree/v1.0, 2019. Last accessed: 27/06/2022.
[13] Y. Kikuta, "BERT pretrained model trained on Japanese Wikipedia articles," https://github.com/yoheikikuta/bert-japanese, 2019. Last accessed: 27/06/2022.
[14] T. Sakaki, S. Mizuki, and N. Gunji, "BERT pre-trained model trained on large-scale Japanese social media corpus," https://github.com/hottolink/hottoSNS-bert, 2019. Last accessed: 27/06/2022.
[15] Y. Kawazoe, D. Shibata, E. Shinohara, E. Aramaki, and K. Ohe, "A clinical specific BERT developed using a huge Japanese clinical text corpus," PLOS ONE, vol.16, no.11, pp.1–11, Nov. 2021.
[16] Y.L. Chen, C.L. Chang, and C.S. Yeh, "Emotion classification of YouTube videos," Decis. Support Syst., vol.101, pp.40–50, Sept. 2017.
[17] Q. Bai, K. Wei, J. Zhou, C. Xiong, Y. Wu, X. Lin, and L. He, "Entity-level sentiment prediction in Danmaku video interaction," J. Supercomput., pp.1–20, 2021.
[18] T. Kudo, "Mecab: Yet another part-of-speech and morphological analyzer," http://mecab.sourceforge.net/, 2005. Last accessed: 27/06/2022.
[19] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," arXiv preprint arXiv:1808.06226, 2018.
[20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," arXiv preprint arXiv:1907.11692, 2019.
[21] Y. Geifman and R. El-Yaniv, "Selective classification for deep neural networks," Proc. 31st International Conference on Neural Information Processing Systems, pp.4885–4894, Dec. 2017.
[22] L. Ziyin, Z.T. Wang, P.P. Liang, R. Salakhutdinov, L.P. Morency, and M. Ueda, "Deep Gamblers: Learning to abstain with portfolio theory," Advances in Neural Information Processing Systems, vol.32, pp.10623–10633, Dec. 2019.
[23] D.P. Kingma and J. Ba, "Adam: A method for stochastic optimization," International Conference on Learning Representations (ICLR), 2015.
[24] S. Moon and N. Okazaki, "Effects and mitigation of out-of-vocabulary in universal language models," J. Information Processing, vol.29, pp.490–503, 2021.
[25] L. Van Der Maaten, "Accelerating T-SNE using tree-based algorithms," J. Mach. Learn Res., vol.15, no.1, pp.3221–3245, Jan. 2014.
[26] H. Suzuki, Y. Miyauchi, K. Akiyama, T. Kajiwara, T. Ninomiya, N. Takemura, Y. Nakashima, and H. Nagahara, "A Japanese dataset for subjective and objective sentiment polarity classification in micro blog domain," Proc. Language Resources and Evaluation Conference, pp.7022–7028, June 2022.
[27] K. Clark, M.T. Luong, Q.V. Le, and C.D. Manning, "ELECTRA: Pre-training text encoders as discriminators rather than generators," International Conference on Learning Representations (ICLR), 2020.

Hiroyoshi Nagao received a B.S. degree in Engineering from Doshisha University in 2022. He is currently a research assistant in the Faculty of Science and Engineering, Doshisha University. His research interests include scientific paper analysis and vision-and-language.

Koshiro Tamura received a B.S. degree in Engineering from Doshisha University in 2021. He is currently a student in the Graduate School of Science and Engineering, Doshisha University. His research interests include natural language processing and blockchain.



Marie Katsurai received a B.S. degree in Engineering, an M.S. degree, and a Ph.D. degree in Information Science and Technology from Hokkaido University in 2010, 2012, and 2014, respectively. She was a Research Fellow of the Japan Society for the Promotion of Science from 2013 to 2015. She joined Doshisha University in 2015 and has been an associate professor since 2021. Her research interests include multimedia information retrieval and academic data analysis. She is a member of the IEICE, IEEE, and ACM.