Personalized Prediction of Offensive News Comments by Considering the Characteristics of Commenters

Teruki Nakahara
Kyushu University
nakahara.teruki.528@s.kyushu-u.ac.jp

Taketoshi Ushiama
Kyushu University
ushiama@design.kyushu-u.ac.jp

Abstract

When reading news articles on social networking services and news sites, readers can view comments marked by other people on these articles. By reading these comments, a reader can understand the public opinion about the news, and it is often helpful to grasp the overall picture of the news. However, these comments often contain offensive language that readers do not prefer to read. This study aims to predict such offensive comments to improve the quality of the experience of the reader while reading comments. By considering the diversity of the readers' values, the proposed method predicts offensive news comments for each reader based on the feedback from a small number of news comments that the reader rated as "offensive" in the past. In addition, we used a machine learning model that considers the characteristics of the commenters to make predictions, independent of the words and topics in news comments. The experimental results of the proposed method show that prediction can be personalized even when the amount of readers' feedback data used in the prediction is limited. In particular, the proposed method, which considers the commenters' characteristics, has a low probability of false detection of offensive comments.

Keywords offensive comments · news · personalization · BERT · user embedding

1 Introduction

In recent years, reading the news on the Internet has become a common practice. According to The Japan Press Research Institute[1], 73.1% of respondents from all age groups read the news on the Internet at least once a week, and among those in their 40s or less, the percentage increases to 90%. In addition to portals and news sites operated by TV stations and newspapers, many people read the news on social networking services (SNSs). In particular, young people in their teens and twenties most frequently use SNSs to access news through the Internet[1].

When reading the news on the Internet, readers can read not only the news but also other people's comments on the news. Yahoo News[1], one of the leading news sites in Japan, has a comments section for each news article. On Twitter[2], one of the most popular SNSs, social-media accounts of TV stations and newspapers post news and links to news articles in the form of tweets. Users can post comments on news stories as replies to these tweets, similar to how they post comments on news sites.

In a survey by Stroud et al.[2], 49% of the people who read news on the Internet said that they also read comments. By reading other people's comments, they can understand the public opinion about the news, obtain an overall picture, and understand it better. However, in online communities, such as news comment sections, readers may find offensive comments, such as those containing extreme or slanderous language.

To solve this problem, methods for predicting comments that are offensive to readers have been proposed[3, 4, 5]. These methods use machine learning models that predict whether a comment is offensive to readers by inputting the comment's text and other information. However, the tendency of comments that readers find offensive differs depending on each reader's values and other factors[6].

---

[1]https://news.yahoo.co.jp/
[2]https://twitter.com/

Table 1: Example of the relationship between comments and standard deviation

SD	News	Comments
0.58	"Internet Explorer" support ends today.	I miss...
0.95	More than 100 subway stations are designated as ballistic missile evacuation facilities for the first time.	Stupid woman. With the current constitution, we'd be destroyed before we could evacuate!
1.26	Prime Minister Kishida "Improve food self-sufficiency and strengthen the international competitiveness of agriculture."	Since the LDP ruined this, the party should be dissolved first.

Figure 1: Distribution of standard deviations in evaluation values

To test the importance of personalization in predicting offensive news comments, we examined the variation in ratings of comments among comment readers. We conducted a survey on the crowdsourcing service CrowdWorks[3] and labeled the news comments. In the survey, 400 news comments posted during the three months from April 1 to June 30, 2022, were presented to 50 users. They were asked to rate the comments on a five-point scale regarding whether they found them offensive {1: Strongly disagree, 2: Disagree, 3: Neither agree nor disagree, 4: Agree, 5: Strongly agree}. In this way, we obtained responses from 50 users per news comment and calculated the standard deviation of the evaluation values for each comment. Examples of the relationship between comments and standard deviations are listed in Table 1. A histogram of the standard deviations of the 400 comments is shown in Figure 1. Figure 1 shows that the standard deviations of the five-point scale are concentrated around 1.0, indicating a variation in the evaluation of comments among comment readers. Therefore, it is essential to personalize the prediction of offensive news comments to consider the values of comment readers.

Personalization methods for predicting offensive comments have been proposed and proven to improve prediction accuracy[7]. Existing methods personalize the prediction of offensive comments by using existing feedback from readers who have previously rated comments as "offensive."

This study aims to generate predictions that are independent of the words and topics in the feedback. Specifically, offensive news comments are predicted by generating embeddings of readers using a machine learning model based on the characteristics of users who posted comments that readers rated as "offensive." In addition, this study used data collected from news tweets by the news TV stations' Twitter accounts and the replies (i.e., comments) to these tweets, and correct answer labels were assigned using a questionnaire.

The contributions of this study can be summarized as follows:

---

[3]https://crowdworks.jp/

- We investigated the variation in the ratings of comments among comment readers and proved the importance of personalization in predicting offensive news comments.
- We personalized the prediction of offensive news comments by generating embeddings of readers based on a small amount of feedback from readers of news comments.
- To generate embeddings of readers, we utilized comments that readers have rated as "offensive" in the past and embeddings of users who have posted comments rated as "offensive" by readers.
- We evaluated the performance of the proposed method by personalizing the prediction of offensive news comments using a dataset that comprises news and comments that have been posted.

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 describes the personalization of offensive news comment predictions addressed in this study. Section 4 describes the proposed method. Section 5 describes the method for generating embeddings of commenters used in the proposed method. Section 6 describes and reports the results of our experiments. Finally, Section 7 summarizes the paper and discusses future research directions.

2 RELATED WORK

2.1 Support for reading news comments

Many studies have been conducted to improve the experience of reading news comments. For example, Ma et al.[8] clustered news comments by estimating the topics of comments using latent Dirichlet allocation (LDA)[9], a topic model. Aker et al.[10] performed clustering of news comments by modeling the similarity among comments using a graph-based method. In these studies, clustering assists readers in viewing news comments by providing them with an overall picture of the news comments. In this study, we improve the quality of the reader experience and promote news comment reading by predicting comments that readers may find offensive, such as those containing extreme content or slanderous remarks.

2.2 Offensive Comments Prediction

Many studies and implementations have been conducted to predict comments that readers may find offensive, such as toxic or insulting comments, which are posted in online communities, such as SNSs. For example, Georgakopoulos et al.[3] used a convolutional neural network (CNN)-based machine learning model to predict toxic comments on editorial discussion pages on Wikipedia. Hessel et al.[5] classified controversial and non-controversial posts on Reddit using models, such as LSTM[11] and BERT[12]. Saveski et al.[13] analyzed the relationship between conversational structure and the toxicity of posts on Twitter. Google Jigsaw provides a system for detecting toxic or insulting comments in Perspective API[4].

2.3 Personalization of Offensive Comment Prediction

Sap et al.[6] described a relationship between the people's identity and beliefs and their toxicity rating in evaluating the toxicity comments. Therefore, it is essential to personalize the prediction of offensive comments by considering differences in the values of comment readers.

Kumar et al.[7] proposed that prediction accuracy can be improved by tuning the parameters of Perspective API provided by Google Jigsaw for each reader, using the feedback of comments that readers have rated as offensive in the past. Generic predictions of comments for personalization can be generated using such feedback, rather than collecting data from specific SNSs or news sites, such as readers' profiles or past postings. However, providing a large amount of feedback is burdensome for readers. Therefore, it is desirable to use only a small amount of feedback data in the prediction.

3 Personalization of Offensive News Comment Prediction

This study addresses the problem of predicting whether a reader of news comments will find the new news comments offensive. For personalization, a general-purpose prediction method is proposed that uses only the feedback of comments rated as "offensive", without using existing data on specific SNSs or news sites, such as readers' profiles or past postings.

We assume that we are given a set of news tweets $NT = \{nt_1, \ldots, nt_{|NT|}\}$, a set of comments $C = \{c_1, \ldots, c_{|C|}\}$, a commenters set $U = \{u_1, \ldots, u_{|U|}\}$, and a readers set $R = \{r_1, \ldots, r_{|R|}\}$. Let $\mathrm{news}(c)$ denote the news tweet to which

Figure 2: Simple prediction model for offensive news comments

the comment c refers and $commenter(c)$ denote the commenter who writes the comment c. Here, $news(c) \in NT$ and $commenter(c) \in U$. Let $P(o \mid c, r)$ denote the probability that the user r judges the comment c to be offensive. In this study, we propose a method to predict this probability using machine learning models. The proposed method considers that the set of news comments rated as "offensive" by reader r is $offensive(r)$. The study aims to predict whether reader r will find a new news comment c offensive.

3.1 Simple Offensive News Comment Prediction Method

In this section, we describe a simple prediction method that serves as a baseline machine learning model for personalizing the prediction of offensive news comments. An overview of the simple prediction model is shown in Figure 2. This model outputs the probability that a given news comment belongs to the "offensive" label based on the vector that concatenates the target and reader vectors. Therefore, the simple prediction model accepts two types of data as input. The first input is the pair of news and comments to be predicted. We vectorize the concatenated news and comment text using BERT, a natural language processing model. The second input is reader ID. In this study, we constructed a Feedback Database, which stores news comments rated as "offensive" by readers alongside those readers' IDs. In the simple prediction model, based on the reader IDs, we retrieve several pairs of news and comment texts from the Feedback database that have been rated as "offensive" in the past. Each output vector obtained by inputting concatenated news and comment texts into BERT is averaged to generate a reader vector of comments.

The aforementioned approach can be formalized as follows: The following equation represents a simple machine learning model that predicts the probability that user r will find comment c offensive:

$$y = f^{sim}(x; \theta^{sim}), \tag{1}$$

where x is a vector representation of the pair of reader r and comment c, y is a scalar value representing the probability value, and θ^{sim} is the set of parameters. The training data set D^{sim} for the simple model is defined as the set of pairs

Figure 3: BERT architecture

(x, y) of input x and output y as follows:

$$D^{sim} = \{(x, y) \mid x = em_c^{sim}(c) \oplus em_r^{sim}(r), c \in C, r \in R, y \in \{1, 0\}\} \tag{2}$$

where x is the concatenation of the embedding vector of comment c and the embedding vector of reader r. Furthermore, $em_c^{sim}(c)$ and $em_r^{sim}(r)$ represent the vectorization of comment c and reader r, respectively, and $\oplus$ represents the vector concatenation operation.

By inputting the text of comment c and the news tweet $news(c)$ it refers into BERT, the embedded vectorization function $em_c^{sim}(c)$ for comment c is obtained and is defined as follows:

$$em_c^{sim}(c) = \text{BERT}(c, news(c)) \tag{3}$$

By contrast, the embedded vectorization function $em_r^{sim}(r)$ of a reader r is defined as follows, using the set $offensive(r)$ of comments deemed offensive by the reader.

$$em_r^{sim}(r) = \sum_{x \in offensive(r)} \frac{\text{BERT}(x, news(x))}{\mid offensive(r) \mid} \tag{4}$$

3.2 BERT for Vectorizing News and Comments

This section describes the vectorization of news and comments using BERT, which can transform input text into contextual vectors by considering word-to-word relationships and conduct various natural language processing tasks, such as sentence classification, with high accuracy. In this study, news and comment texts are combined in the following format, and the text divided into tokens is used as input to BERT:

[CLS] news text [SEP] comment text [SEP],

where [CLS] is a special symbol that is added before each input example, and [SEP] is a special separator token. Using the self-attention mechanism in the transformer encoder[14], BERT outputs a vector that considers the relationship between words in the input data. Figure 3 shows the architecture of BERT used in this study, where E is the input embedding, Trm is the transformer encoder, and T is the output from the last hidden layer of the transformer encoder. The output vector of the [CLS] token from the last hidden layer of the transformer encoder is used as the vector of news and comments for the method of predicting offensive news comments described in Chapters 3 and 4.

3.3 Problems with Simple Offensive News Comment Prediction Model

This section describes the problems with the simple prediction model stated in Section 3.1. This method uses the feedback of news comments that readers rated as "offensive" in the past to personalize the prediction. However,

5

Figure 4: Overview of the proposed method

providing a large amount of feedback is burdensome for readers. Therefore, it is desirable to use only a small amount of feedback data for prediction. However, if the amount of feedback data used for prediction is limited, the prediction results may be strongly influenced by the words and topics in the feedback. For example, we assume that only comments on political news are included in the feedback. In that case, correct predictions for comments on political news may be possible, although correct predictions for comments on sports news may be challenging. Similarly, it may be difficult to predict comments on the latest news that is not included in the feedback.

4 Proposed Method

This section describes our proposed method for predicting offensive news comments that considers the commenter's characteristics.

4.1 Overview of the Proposed Method

Figure 4 shows an overview of the proposed method. The proposed method outputs the probability that the news comment to be predicted belongs to the "offensive" label based on a vector that concatenates the target and reader vectors. In addition, the proposed method considers the characteristics of the commenters in the prediction. In Figure 4, the characteristics of the commenters are used when generating the target and readers vector.

To personalize the predictions, we use feedback from news comments that readers have rated as "offensive" in the past. The simple prediction model described in Section 3.1 is expected to be strongly influenced by the words and topics in the feedback. The proposed method, a machine learning model using the features of commenters, considers the features of the user who posted the news comment that the reader rated as "offensive" and the features of the user who posted the news comment to be predicted. This method is expected to result in predictions independent of the words and topics included in the feedback.

6

The aforementioned approach can be formalized as follows: The proposed machine learning model that predicts the probability that user r finds comment c to be offensive is represented by the following equation:

$$y = f^{pro}(x; \theta^{pro}), \tag{5}$$

where x is a vector representation of the pair of reader r and comment c, y is a scalar value representing the probability value, and θ^{pro} is the set of parameters. The training data set D^{pro} for the proposed model is defined as the set of pairs (x, y) of input x and output y as follows:

$$D^{pro} = \{(x, y) \mid x = em_c^{pro}(c) \oplus em_r^{pro}(r), c \in C, r \in R, y \in \{1, 0\}\} \tag{6}$$

where x is the concatenation of the embedding vector of comment c and the embedding vector of reader r. In addition, $em_c^{pro}(c)$ and $em_r^{pro}(r)$ represent the proposed vectorization of comment c and reader r, respectively.

4.2 Generating Target Vectors for the Prediction

To generate the target vectors, pairs of news and comments to be predicted are converted into vectors using BERT. Next, the users who posted the news comments to be predicted are vectorized by the commenter encoder model described in Section 5. Finally, we generate the target vector by concatenating the vector of news and comments with the vector of the commenter. The vectorization of the target comment c in the proposed method can be defined as follows:

$$em_c^{pro}(c) = em_c^{sim}(c) \oplus enc(commenter(c)), \tag{7}$$

where $enc(u)$ denotes the function to vectorize the commenter u with the proposed commenter encoder model.

4.3 Generating Reader Vectors

To generate reader vectors, the reader ID is first entered. Then, based on the reader ID, we retrieve the texts of pairs of news and comments that the target reader has rated as "offensive" in the past and user IDs of the commenters who posted the comments from the Feedback Database. Each output vector obtained by inputting the acquired news and comments to BERT is averaged. This produces a vector of news and comments that the reader has rated as "offensive" in the past. Next, the users who posted comments that the reader rated as "offensive" in the past are vectorized for each user using the commenter encoder model, and each vector is averaged. Finally, a reader vector is generated by concatenating the vectors of news and comments rated as "offensive" and the vector of the user who posted the comment that was rated as "offensive." The proposed vectorization function of a reader r is defined as follows:

$$em_r^{pro}(r) = em_r^{sim}(r) \oplus \sum_{x \in offensive(r)} \frac{enc(commenter(x))}{\mid offensive(r) \mid} \tag{8}$$

5 Commenter Encoder Model

A machine learning model was proposed to vectorize the characteristics of commenters using the pairs of news and comment text; it has been proven that the vector adequately represents the characteristics of commenters[15]. The proposed method uses the same method to vectorize the features of commenters for prediction through a machine learning model that considers the features of commenters.

5.1 Method for Vectorizing Characteristics of Commenters

This section describes the commenter encoder model used to vectorize the characteristics of commenters. Figure 5 shows an overview of this model. First, we construct a database that stores past news and comments along with the commenter IDs, which we refer to as the News and Comment Database. The commenter encoder model retrieves multiple texts of pairs of comments posted in the past by the commenter and the news from the News and Comment Database by entering the commenter ID. In this study, the number of retrieved cases was set to five. The retrieved text of pairs of news and comments is converted into a vector using BERT, and commenters are classified using the fully connected layer. By predicting the commenters, we can embed the characteristics of the commenters into the vectors in the hidden layer of the prediction model. In this method, the BERT layer in predicting commenters is used to generate a vector of commenters, and the average of the output of the BERT layer is used as the commenter vector for predicting offensive news comments. When N pairs of news and comments are obtained from a commenter u, let $\{d_1, d_2, \ldots, d_N\}$ be the text set of pairs of news and comments and $g(d_k)$ be the output obtained from text d_k by the BERT layer. The following formula defines the function to vectorize the commenter u.

Figure 5: Overview of the commenter encoder model

$$enc(u) = \frac{1}{N}\sum_{k=1}^{N} g(d_k) \tag{9}$$

5.2 Definition of the Training Dataset for the Commenter Encoder Model

The commenter encoder model predicts the commenter from the input texts of news and comment pairs and embeds the commenter's features into the vector in the hidden layer. Therefore, to incorporate the commenter encoder model into the proposed method, we first train the commenter prediction in the commenter encoder model. Given a pair of news and comment (nt, c), we consider predicting the commenter. Let $P(u \mid nt, c)$ denote the probability that the commenter is u. We assume that $\sum_{u \in U} P(u \mid nt, c) = 1$. The proposed method uses a machine learning model that predicts $P(u \mid nt, c)$ to generate an embedding of the commenter. Let (x, y) denote the training data used to train the prediction model, where x is the input data, and y is the ground truth for that data. We define the training dataset D^{com} of the commenter encoder model as follows:

$$D^{com} = \{(x, y) \mid x = BERT(c, news(c)), y = commenter(c), c \in C, y \in U\} \tag{10}$$

6 Experiments

6.1 Data and Settings

6.1.1 Commenter Encoder Model

We constructed the News and Comment Database, which comprises news tweets posted by NHK News (@nhk_news)[4] from November 11, 2021 to March 31, 2022, users' replies to the news, and user IDs of the users who replied to the news, which are referred to as news, comments, and commenter IDs, respectively. These were obtained using the Twitter API.

The commenter encoder model used to vectorize the characteristics of commenters accepts pairs of news and comments as input and classifies commenters. The features of the commenters are embedded into the vectors in the hidden layer

---

[4]https://twitter.com/nhk_news

Table 2: Total number of each label obtained in the survey

Label	Total
not offensive	70,800
offensive	29,200

of the prediction process. We trained the commenter encoder model on 515 users with more than 50 comments in the News and Comment Database for use in the proposed method. The BERT model was trained by fine-tuning a pre-trained BERT model[5] using Wikipedia, which was developed by Inui Lab at Tohoku University. We randomly obtained 50 pairs of news and comments from users; we used 40 of each user's posts as training data, five as validation data, and the remaining five as test data. The input to the model for one datum is a single pair of news and comment, and the output is a 515-dimensional probability vector.

6.1.2 Model for Offensive News Comment Prediction

We labeled news comments by conducting a survey on the crowdsourcing service CrowdWorks. The total number of subjects was 250, divided into five groups of 50 each. Each group was presented with 400 news comments posted during the three months from April 1 to June 30, 2022, and asked to respond on a 5-point scale regarding whether they found each news comment offensive {1: Strongly disagree, 2: Disagree, 3: Neither agree nor disagree, 4: Agree, 5: Strongly agree}. The news and comments presented to the subjects were the same in each group. Labels for 400 news comments were obtained per subject by assigning two labels to news comments with responses of 4 or 5 as "offensive" and others as "not offensive." In other words, the total number of labels obtained was 250×400. The total number of each label is listed in Table 2.

We constructed the Feedback Database, which contains the pairs of news and comments rated as "offensive" during the two weeks from April 1 to April 14, 2022, the commenter ID of the user who posted each comment, and the reader ID of the user who made the rating. However, the Feedback Database does not store news and comments rated as "not offensive." Therefore, the proposed method generates a reader vector by accepting the reader IDs as inputs and acquiring the pairs of news and comments rated as "offensive" and the commenter IDs from the Feedback Database. Finally, we use the generated reader vectors to personalize the prediction.

The input training data are the pairs of news and comments to be predicted as well as the reader IDs, and the correct answer labels are the two labels of "offensive" or "not offensive" provided by the readers. The news comments from April 15 to June 9, 2022, from June 10 to June 20, 2022, and from June 21 to June 30, 2022, were used as the training data, validation data, and test data, respectively, in chronological order.

We trained the model to predict offensive news comments on 192 comment readers for whom we had obtained at least five feedbacks from the Feedback Database. For the BERT model for vectorizing news and comments, we used a pre-trained BERT model based on Japanese Wikipedia by Inui Lab at Tohoku University and conducted fine-tuning. The two BERT layers shown in Figure 4 are trained without sharing weights. Because the act of providing a large amount of feedback is burdensome for readers, it is desirable to use only a small amount of feedback data from readers for personalization. Therefore, we limited the personalization to five news comments that were rated as "offensive" in the past when the reader ID was entered.

6.1.3 Pre-processing

The following preprocessing was applied to the news texts used in this study:

- Remove URLs
- Remove symbols
- Remove hashtags (e.g., #nhk_news)

In addition, the following preprocessing was applied to the comment texts in this study:

- Remove URLs
- Remove symbols
- Remove emojis
- Remove mentions (@user ID)

---

[5]https://github.com/cl-tohoku/bert-japanese

9

Figure 6: PR curves

Table 3: AUCs of the PR curves

Model	AUC
Simple Prediction Model	0.334
Proposed Model	**0.339**
Model without Personalization	0.223

6.1.4 Description of Comparison Models

To evaluate and compare the results of the offensive news comment prediction, we compared the prediction results of three different models in this experiment. The details of each model are as follows:

Simple Prediction Model
As shown in Figure 2, we generate a reader vector using multiple feedbacks of news comments that readers have rated as "offensive" in the past and personalize the prediction of offensive news comments.

Proposed Model
As shown in Figure 4, we personalize the prediction of offensive news comments by incorporating a structure that utilizes the characteristics of commenters into a simple prediction model.

Model without Personalization
We predict offensive news comments by using only the news comment to be predicted. Because the reader vectors are not used for prediction, the prediction is not personalized.

6.2 Results and Discussion

6.2.1 Comparison by the precision-recall (PR) curve

Figure 6 shows a PR curve based on the prediction results of the three models on the test data. The PR curve is a plot of the relationship between precision and recall while changing the threshold for the prediction probability output by the prediction model. By comparing the precision for the same recall, the performance of the models can be compared without manually setting a threshold. PR curves are proved to be suitable for evaluating classifications with a biased distribution of labels, as demonstrated in this experiment. Table 3 summarizes the AUC values for the PR curves.

First, we examined the validity of the personalization of the predictions. Figure 6 and Table 3 show that the prediction performance of the models with personalization (proposed model and simple prediction model) significantly exceeded

Table 4: Metrics for evaluating the simple prediction model at each threshold

Threshold	Accuracy	Recall	Precision	F-measure
0.9	0.838	0.003	0.333	0.005
0.8	0.838	0.052	0.500	0.095
0.7	0.833	0.146	0.448	0.220
0.6	0.817	0.282	0.404	0.332
0.5	0.779	0.431	0.351	0.387
0.4	0.707	0.607	0.300	0.402
0.3	0.582	0.767	0.246	0.373
0.2	0.416	0.899	0.204	0.333
0.1	0.232	0.986	0.172	0.293

Table 5: Metrics for evaluating the proposed model at each threshold

Threshold	Accuracy	Recall	Precision	F-measure
0.9	0.839	0.012	0.667	0.023
0.8	0.839	0.064	0.532	0.114
0.7	0.837	0.145	0.491	0.224
0.6	0.824	0.225	0.419	0.293
0.5	0.804	0.329	0.380	0.352
0.4	0.763	0.450	0.330	0.381
0.3	0.698	0.607	0.292	0.394
0.2	0.576	0.763	0.242	0.368
0.1	0.357	0.938	0.193	0.321

Table 6: Precision@k

Model	k=1	k=3	k=5	k=10
Simple Prediction Model	0.378	0.402	0.430	0.472
Proposed Model	**0.450**	**0.484**	**0.445**	**0.481**

that of the model without personalization. This result indicates that personalization of predictions is effective in predicting offensive comments.

Next, we compared the performance of the proposed model with that of the simple prediction model. Figure 6 shows that the proposed model outperformed the simple prediction model in the recall range from 0.0 to 0.2. Moreover, the proposed model achieved lower or equal precision in the recall range from 0.2 to 1.0. These results indicate that the proposed method, which considers the characteristics of the commenter, is effective in situations with low recall for comments with high predictive probability and a low probability of false detection of offensive comments. Table 4 lists the metrics for evaluating the simple prediction model when the threshold used to determine a news comment as "offensive" or "not offensive" is manually changed, and Table 5 shows the metrics for evaluating the proposed model.

6.2.2 Comparison by Precision@k

The predicted probabilities for the test data output by the prediction model were placed in descending order. The percentage of correct labels when the top k data were determined as "offensive" (Precision@k) was calculated for each reader. The model's performance was compared by averaging the Precision@k for each reader. Readers with fewer than k comments rated as "offensive" in the test data were excluded. The values of Precision@k for k=1, 3, 5, and 10 are listed in Table 6. Based on the total number of each label shown in Table 2, the chance level of Precision@k is approximately 0.292 (29,200/100,000). Table 6 shows that the proposed model outperforms the simple prediction model in terms of prediction performance, with values greater than the chance level observed for both models. This result indicates that the prediction can be personalized even when the amount of readers' feedback data used in the prediction is small. Furthermore, the personalization of the proposed model, which considers the characteristics of the commenter, is less affected by the words and topics included in the feedback, which may have led to the improvement in prediction performance.

In addition, for small values of k, such as one or three, the proposed model outperforms the simple prediction model by a significant margin. As explained in Section 6.2.1, this is because the proposed model, which considers the characteristics of the commenter, is effective when targeting comments with a high predicted probability.

7 Conclusion

In this study, we attempted to personalize the prediction of offensive news comments based on a small amount of feedback from news comment readers who rated news comments as "offensive" in the past. To generate predictions independent of the words and topics in the feedback, we proposed a machine learning model that considers the characteristics of the commenter. By analyzing the variations in the ratings of news comments, we found that variation exists in the ratings of comments among readers, indicating the importance of personalization in predicting offensive news comments. Furthermore, the experimental results showed that personalization of predictions is possible even when the amount of readers' feedback data used in the prediction is small. In particular, the proposed method, which

considers the characteristics of the commenter, has a low probability of false detection of offensive comments. In the future, we will attempt to improve the performance of the proposed method to accurately predict offensive news comments. The prediction model's current performance is insufficient for practical use, and the model needs to be improved. We are also investigating the effect of the amount of feedback data on the personalization of predictions by examining the change in prediction performance when the amount of readers' feedback data used in the prediction is varied.

Acknowledgments

References

[1] The Japan Press Research Institute. 14th national public opinion survey on media, 2021.

[2] Natalie Jomini Stroud, Emily Van Duyn, and Cynthia Peacock. News commenters and news comment readers. *Engaging News Project*, pages 1–21, 2016.

[3] Spiros V. Georgakopoulos, Sotiris K. Tasoulis, Aristidis G. Vrahatis, and Vassilis P. Plagianakos. Convolutional neural networks for toxic comment classification. Number 35 in In Proceedings of the 10th Hellenic Conference on Artificial Intelligence, pages 1–6, 2018.

[4] Google Jigsaw. Perspective api, 2017.

[5] Jack Hessel and Lillian Lee. Something's brewing! early prediction of controversy-causing posts from discussion features. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1648–1659, 2019.

[6] Maarten Sap, Swabha Swayamdipta, Laura Vianna, Xuhui Zhou, Yejin Choi, and Noah A. Smith. Annotators with attitudes: How annotator beliefs and identities bias toxic language detection. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics, pages 5884–5906, 2022.

[7] Deepak Kumar, Patrick Gage Kelley, Sunny Consolvo, Joshua Mason, Elie Bursztein, Zakir Durumeric, Kurt Thomas, and Michael Bailey. Designing toxic content classification for a diversity of perspectives. In Proceedings of the Seventeenth Symposium on Usable Privacy and Security, pages 299–318, 2021.

[8] Zongyang Ma, Aixin Sun, Quan Yuan, and Gao Cong. Topic-driven reader comments summarization. In Proceedings of the 21st ACM international conference on Information and knowledge management, pages 265–274, 2012.

[9] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[10] Ahmet Aker, Emina Kurtic, A. R. Balamurali, Monica Paramita, Emma Barker, Mark Hepple, and Rob Gaizauskas. A graph-based approach to topic clustering for online comments to news. In Proceedings of the 38th European Conference on IR Research, pages 15–29, 2016.

[11] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1977.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 4171–4186, 2019.

[13] Martin Saveski, Brandon Roy, and Deb Roy. The structure of toxic conversations on twitter. In Proceedings of the Web Conference, pages 1086–1097, 2021.

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Proceedings of the 31st Conference on Neural Information Processing Systems, pages 6000–6010, 2017.

[15] Teruki Nakahara and Taketoshi Ushiama. Generating user embedding based on reactions to news for reading news comments support. *IPSJ Transactions on Databases(TOD)*, 15(3):99–110, 2022.