Relationship Between Personality Patterns and Harmfulness:

Analysis and Prediction Based on Sentence Embedding

Kazuyuki Matsumoto, Tokushima University, Japan*
Ryota Kishima, Tokushima University, Japan
Seiji Tsuchiya, Doshisha University, Japan
Tomoki Hirobayashi, Yamada Denken Co., Ltd., Japan
Minoru Yoshida, Tokushima University, Japan
Kenji Kita, Tokushima University, Japan

ABSTRACT

This paper hypothesizes that harmful utterances need to be judged in the context of whole sentences, and the authors extract features of harmful expressions using a general-purpose language model. Based on the extracted features, the authors propose a method to predict the presence or absence of harmful categories. In addition, the authors believe that it is possible to analyze users who incite others by combining this method with research on analyzing the personality of the speaker from statements on social networking sites. The results confirmed that the proposed method can judge the possibility of harmful comments with higher accuracy than simple dictionary-based models or models using a distributed representation of words. The relationship between personality patterns and harmful expressions was also confirmed by an analysis based on a harmful judgment model.

KEYWORDS

Deep Neural Networks, Harmful Expression, Internet Flaming Detection, MBTI, Personality, Sentence Embeddings, Text Classification

1. INTRODUCTION

There are various risks associated with posting provocative or offensive messages online (also known as “Internet flaming”). The risk of personal information being leaked can lead to inquiries and harassment via email and phone; if the situation becomes serious, it can lead to the breakdown of relationships and affect unrelated people. To avoid such risks, it is important to prepare measures to prevent Internet flaming.

There are many causes of online flame wars. In particular, microblogs such as Twitter allow users to post easily; therefore, even extreme or inappropriate content that gives a bad impression is often posted without careful thought. For example, some posts share annoying behavior or pranks or

DOI: 10.4018/IJITWE.298654

*Corresponding Author

boasts about criminal activities. These posters have no idea that what they post on social networking sites (SNS) will be seen by many people. All of these have the potential to cause a social media storm.

According to statistical studies, there is a slight tendency for flaming participants to be complicit in online flames according to their gender, age, and annual income (Tanaka & Yamaguchi, 2016). However, anyone who uses SNS can be a participant or a victim of online flaming.

In addition, artificial intelligence (AI) chatbots, which have been developing rapidly in recent years, can generate natural and fluent speech as if spoken by a human. The language models used to generate this speech are often trained using large-scale text data collected from the Internet. Therefore, there is a problem that inappropriate expressions in the text data can be reflected in the speech generation, resulting in harmful statements (Fuchs, 2018).

The causes and forms of all recent Internet flames are diverse, and it is difficult to detect them using only the text posted as a clue. There are two problems: 1) the text may contain images or videos that may upset users, and 2) most of the posts in cases of Internet flaming are deleted immediately after the reaction caused.

First, it is not impossible to analyze the meaning of images and videos, but more complicated processing is required. Second, it is difficult to collect data on a large scale. Therefore, here, we focus on harmful expressions such as swear words and discriminatory statements, which are less dependent on a specific time or speaker and may be repeated by many people.

It is not always the case that harmful expressions are included in comments that cause flames. If a tweet contains abusive language, depending on the context, it may not be seen as an example of Internet flaming. For this reason, it is difficult to predict whether a tweet will become an Internet flame. However, if a tweet is judged to be harmful, there will be many people who will misunderstand the tweet. In other words, it is possible to prevent flaming by informing the speaker of the possibility that a harmful text may trigger a flaming incident.

Here, the utterances are collected to be analyzed based on the harmful expression dictionary defined in Matsumoto et al.'s study (Matsumoto et al., 2018). It is created that a corpus of harmful expressions by assigning harmful categories to these sentences. The sentences are then vectorized using a pre-trained language model such as the bidirectional encoder representations from transformers (BERT) model, and a deep neural network is used to train the harmful category classification model and evaluate its accuracy. In addition, by analyzing the relationship between the user's personality pattern, which can be inferred from the content of the utterance, and the harmfulness of the utterance, it is thought that it can be useful for predicting flaming before it occurs. Here, we focus on the similarity between harmful comments and personality patterns and analyze what kind of personality is likely to tweet harmful comments.

The remainder of the paper consists of the following. Section 2 describes the related research and Section 3 the proposed method. Section 4 discusses the results of the classification experiments and their similarity to personality patterns, and Section 5 summarizes the results.

2. RELATED WORKS

2.1 Study of Internet Flaming

Many studies have been conducted on detecting Internet flaming (Ozawa et al., 2016; Steinberger et al., 2017; Lingam et al., 2017; Yoshida et al., 2014; Yoshida et al., 2016; Babakov et al., 2021; Ball-Burack et al., 2021). One study detected flaming based on the emotional polarity of words (Ozawa et al., 2016). Another study determined which news topics are likely to cause flaming (Steinberger et al., 2017), rapidly increase negative comments and detecting the moment when a flaming occurs. In (Yoshida et al., 2014), a sentiment analysis based on naïve Bayesian classification was used to detect flaming from comments posted by users on a private company's SNS. These studies are useful techniques for organizations such as companies, but they do not have many advantages for individual users who are exposed to the risk of flaming. In addition, (Babakov et al., 2021) proposed a method to define sensitive topics that may produce inappropriate and harmful messages, collect datasets, and label them for appropriateness. While their method focused on topics, our method differs in that it targets topics without limitations. This section present our research on the classification of potentially inflammatory expressions.

Matsumoto et al. (Matsumoto et al., 2018) aimed to create a system that checked whether a sentence contained potentially inflammatory expressions and prompted the user to correct them. They created a dictionary of harmful expressions, a dictionary of general expressions, and trained support vector machines (SVMs) on the two dictionaries to determine harmful expressions. They used word distributed representation as a word feature.

The SVM is more useful for judging harmful expressions than simple matching of the harmful expression dictionary. However, there were some cases in which the SVM was not able to make correct judgments on a word-by-word basis because of the connection between the preceding and following words. For example, in the sentence, "I was a *kimo-ota* too," the word "*kimo-ota*" was judged to be harmful, but the sentence as a whole was self-deprecating and did not hurt others, so the possibility of flaming was considered low.

In many existing studies on swear words, the ease of flaming was estimated based on the harmfulness of the word itself. However, one problem with this approach is that it can also be used to judge the likelihood of flaming, even when harmful expressions are included, such as self-mockery. For this reason, it is believed that word-by-word judgment is insufficient and that it is necessary to extract meaning from the sentence itself.

2.2 Detecting Internet Cyberbullying

To prevent cyberbullying on electronic bulletin boards, there is a system that automatically analyzes the posted contents and deletes or reports it to the administrator if the posts are judged to be harmful. Research on cyberbullying has used clustering methods to detect the category of cyberbullying to which a user belongs (Sarna & Bhatia, 2020). Clustering methods such as neural network models to judge cyberbullying (Bozyiğit et al., 2020), and machine learning (ML) methods such as (Çürük et al., 2018; Sani et al., 2017).

In all of these cases, they prepared a data set that was pre-labeled based on positive and negative examples of cyberbullying or categories of cyberbullying from data collected manually and evaluated the classification accuracy using a ML model. Since swear words are an important factor in the identification of cyberbullying, the key point is how to appropriately create and augment the list of swear words. In the section that follows, it is introduced that the researches on a method for collecting swear words from bulletin board sites.

Ishisaka et al. (Ishisaka et al., 2010) extracted swear words and phrases from the large anonymous electronic bulletin board site "2channel" and constructed a dictionary of swear words and phrases. They defined a swear word as a word or phrase that was directly insulting or defamatory to a specific person. Using word n-grams, they proposed a method for extracting swear words from the surrounding word sequences. As an evaluation set, they manually prepared a set of 378 swear words and 382 non-swear words from 2channel and evaluated the number of new swear words not registered in the dictionary. They set a threshold value for the word association probability of the word n-gram extraction, and when the threshold value was 0.9 or 0.8, the fitting rate was 1.0. In this case, instead of perfectly extracting swear words, the number of swear words that could be extracted was three, and the reproduction rate was approximately 0.3, indicating a problem in detection performance.

Ptaszynski et al. (Ptaszynski et al., 2017) proposed a method for detecting cyberbullying posts based on morphosemantic patterns. As a result of comparative experiments using a variety of features, the detection accuracy was up to 90% in some cases, but because it was a threshold-based method, the reproduction rate tended to be low, as in the study by Ishisaka et al. (Ishisaka et al., 2010).

Choi et al. (2021) used text mining techniques to detect cyberbullying in social networks. In their complementary method, they calculated the Losada ratio, which is the ratio of comments with positive content to those with negative content, and analyzed the relationships among users to check the influence of core users. In practice, the proposed method contributes to the management of online communities and the reduction of cyber bullying.

Perera et al. (2021) considered the main characteristics of cyberbullying, such as intent to harm individuals, repetition, passage of time, abusive curl language, or hate speech using supervised machine learning. They proposed an automatic detection and prevention system for cyberbullying. Their system used support vector machines and logistic regression to detect cyberbullying texts and topics related to cyberbullying, such as racism, sexuality, physical averages, and taunts. They presented a new hypothesis of cyberbullying detection in which text messages and their language status and usage change over time. Most previous studies have considered cyberbullying to be stupid, ugly, and ridiculous to others, but these words are not always cyberbullying today. They use more extreme language when they want to deliberately cause psychiatric damage to an individual. In addition to conventional feature extraction methods such as term frequency-inverse document frequency (TF-IDF), the accuracy of cyberbullying detection systems has improved by using n-grams and derogatory expressions as features along with sentiment analysis.

There are other studies on detecting cyberbullying; Fortunatus et al.(2020) used a rule-based method with an extended dictionary to achieve high accuracy in detecting cyberbullying. In addition, Bozyiğit et al. (2021) analyzed the features that are closely related to cyberbullying using the chi-square test as a method to automatically detect cyberbullying content on social networks. Several characteristic trends were observed. For example, it became clear that users with a large number of followers tended to be reluctant to engage in cyberbullying content. It has been reported that a machine learning algorithm was used to predict net tightening by social network features and text features and showed high performance.

López-Vizcaíno et al. (2021) explored various time-sensitive approaches to detecting cyberbullying in social networks. Their method was based on a supervised learning method with two different early detection models. To evaluate the validity of the proposed method, they used real-world datasets to perform time-sensitive experimental evaluations that penalized detection delays, resulting in a maximum baseline detection model. It improved by 42%.

2.3 Detection of Hate Speech and Other Harmful Comments on the Internet

Karayiğit et al. (Karayiğit et al., 2021) proposed a method for detecting abusive comments on Instagram in Turkey. In their method, a new dataset called Abusive Turkish Comments was proposed and created to detect abusive comments on Instagram written in Turkish. The dataset consisted of 10,528 abusive comments and 19,826 non-abusive comments posted on tabloid and sports newspaper accounts. This was the first public dataset dedicated to the detection of abusive comments in Turkish. The sentiment annotations were assigned polarity for each comment unit. Other studies that created datasets included (Omar et al., 2020).

Watanabe et al. (Watanabe et al., 2018) proposed a method for detecting hateful expressions on Twitter. Their method was based on an ML algorithm that used unigrams and patterns automatically collected from a training set. They experimented with a test set of 2,010 tweets to determine whether a tweet was offensive. The results showed that the accuracy of detecting offensive tweets reached 87.4% (binary classification), and the accuracy of detecting whether a tweet was hateful, offensive, or clean reached 78.4% (ternary classification).

Kapli et al. (2020) used deep neural networks to detect hate speech posted on social media. In their approach, they applied convolutional neural networks and recurrent neural networks and proposed a model to classify datasets based on multiple tag systems on a single network. For example, a dataset whose posts are classified by offensive/non-offensive labels, or a dataset classified by harassment/neutral. The input text features for each task were trained using a neural network (shared-private multi-task learning; SP-MTL). For each task, the final layer produced an output using a softmax function. Their method was efficient in that it did not unify the classification systems for hate speech, but considered them as separate tasks, such that the user was not aware of the differences in classification systems. In our study, we assumed three types of speech (rhetoric, hate, and criminal speech) and four other classes from the viewpoint of flame detection, and the features used as input to the neural network were different: Kapli et al. (2020) used word and character embeddings, whereas we used embedding representations from the BERT's pre-trained model. Furthermore, our objectives differed in that we investigated the relationship between a user's personality based on the flame detection model.

The hate expressions in the datasets used in these studies were likely to vary depending on the attributes of the people they represented and the hate targets. Therefore, if a single dictionary was constructed, it might not be possible to detect hate speech in different communities. Here, we focus only on posts in Japanese on SNSs. In addition, this study requires a more detailed categorization than the aforementioned studies because it deals not only with hate speech but also with offensive speech, criminal threats, etc. Furthermore, because of the wide variety of expressions, we extract features from sentences using a language model that can be used flexibly without relying on existing dictionaries.

2.4 Study on Personality Analysis

The personality analysis used in this study was the Myers–Briggs Type Indicator (MBTI), which is a psychological self-report assessment method. This method has 16 personality types that are roughly divided into four dichotomies: extraverted-introverted, sensory-intuitive, thinking-emotional, and judgmental-cognitive. The Big Five personality traits, which are more commonly used than the MBTI, are similar to the MBTI, and there are correlations between the factors defined by each method.

Several studies have estimated personality patterns based on MBTI from linguistic information (Amirhosseini & Kazemian, 2020; Keh & Cheng, 2019; Yamada et al., 2019; Khan et al., 2020; Mushtaq et al., 2020; Choong & Varahan, 2021; Kishima et al., 2021). There is also a study using the Big Five (Salsabila & Setiawan, 2021). In addition, Ezpeleta et al. (2018) used personality traits and emotional information of MBTI for spam detection in short message service (SMS). Jabos (2019) estimated the personality of characters in fictional works based on word sentiment analysis.

Many studies have been conducted to analyze the author's personality based on the content of a text. However, there have not been many studies that have applied the results of these analyses, and we believe that our analysis can contribute to the prevention of Internet flaming through the successful use of personality analysis methods.

3. PROPOSED METHOD

3.1 Fine-tuning Based on BERT Pre-Trained Model

Here, a method is proposed for making binary judgments of harmfulness or harmlessness using word distributed representations based on BERT (Devlin et al., 2018) as feature values and multiple ML models. In contrast to conventional methods that use word distributed representations as a feature value to determine whether a word is harmful or not, our method determines whether a word is harmful in a sentence. It is believed that interactive learning methods based on BERT and other masked language models are effective in capturing the difference in the meaning of words in a sentence. There are also bidirectional neural networks, such as bidirectional long-short term memory (Gers et al., 2000). However, one of the characteristics of long-short term memories is that they are prone to overtraining depending on the problem being handled.

There was also an improved version using convolutional neural networks (CNNs) for machine translation (Gehring et al., 2017). However, in the case of CNNs, it was difficult to account for long-range dependencies. BERT was a model that solved the long-range dependencies with the attention mechanism and successfully captured semantic relations between words and contexts by training a bi-directionally larger language corpus with a mask language model and a next sentence prediction task. Fig. 1 shows the basic structure of Transformer Encoders. BERT uses a bi-directional transformer encoder to train Masked Language Model (MLM) and Next Sentence Prediction (NSP) tasks in pre-training. In general, BERT uses fine-tuned models that have been pre-trained on large training datasets. The model with 12-layer transformer encoders is $\text{BERT}_{\text{Base}}$, with a hidden layer dimensionality of 768; $\text{BERT}_{\text{Large}}$, with 24-layer transformer encoders, has a hidden layer dimensionality of 1024. In this study, $\text{BERT}_{\text{Base}}$ is used as the pre-trained model.

Here, the authors also built a model for determining the harmfulness of tweet sentences by fine-tuning using a pre-trained model based on a Japanese corpus. For tasks such as sentence clustering, the distributed representation vector of tokens corresponding to CLS is often used as a feature (see Fig. 2).

Improved versions of BERT include DistilBERT (Sanh et al., 2019), which successfully reduces the model size by reducing the number of parameters and Sentence BERT (Reimers & Gurevych, 2019), which specializes in representing sentence similarity. Both of these models are applications of the BERT mechanism but have slightly different uses and purposes. In this research, the authors believed that there was no essential difference between BERT and DistilBERT, and the authors also built a network that used the BERT vector of CLS tokens (BERT-CLS), the mean vector (BERT-AVG), and DistilBERT (DBERT) as input features.

Using the BERT's pre-trained model, it is possible to extract generic feature representations from short sentences, considering the context. Indeed, there are many examples of improving the accuracy of text classification tasks by applying transfer learning to a set of labeled documents according to the task, based on the features obtained from the pre-trained model. Zhao et al. (2021) proposed a method via the parameter transfer and attention sharing mechanism (PTASM), using a model built using a pre-training language model of training data, such as BERT, in a cross-domain dataset of Amazon reviews. They achieved cutting-edge results.

The bag-of-words vector was also a conventional feature used. In the case of the bag of words, the number of dimensions increased depending on the number of words that appeared; therefore, to fix the number of dimensions, there was a method to fix the number of dimensions by using singular value decomposition (SVD). Here, it is also used BoW-SVD as a feature.

3.2 Harmful Expression Corpus

To detect harmful sentences, it was not sufficient to collect a corpus containing harmful expressions. It was necessary to prepare harmless examples (negative examples) of statements that were harmless, though they contained harmful expressions.

For this purpose, it is collected that tweet sentences according to the following criteria: The harmful expressions dictionary used for collecting tweets contained 859 words of harmful expressions in Japanese.

1. Collect tweet sentences included in the harmful expressions dictionary using Twitter API.
2. Collect harmful sentences from replies to tweets of users who have been burned.
3. Collect examples of statements from websites that summarize incidents such as criminal threats.

For 1 and 2, the classification tags of "abusive or violent speech, (Violence)," "hate speech, (Hate)" and "other" were assigned manually. For 3, a "crime" tag was assigned.

In this way, it was defined that the set of tagged tweets as a corpus of harmful expressions. We defined the set of tweets tagged with "other" as the corpus of harmful expressions.

Table 1 presents an overview of the corpus of harmful expressions constructed here. MeCab[1], a Japanese morphological analyzer, was used for word sharing, and the MeCab-Juman dictionary was used for morphological analysis.

Figure 1. Architecture of transformer encoders

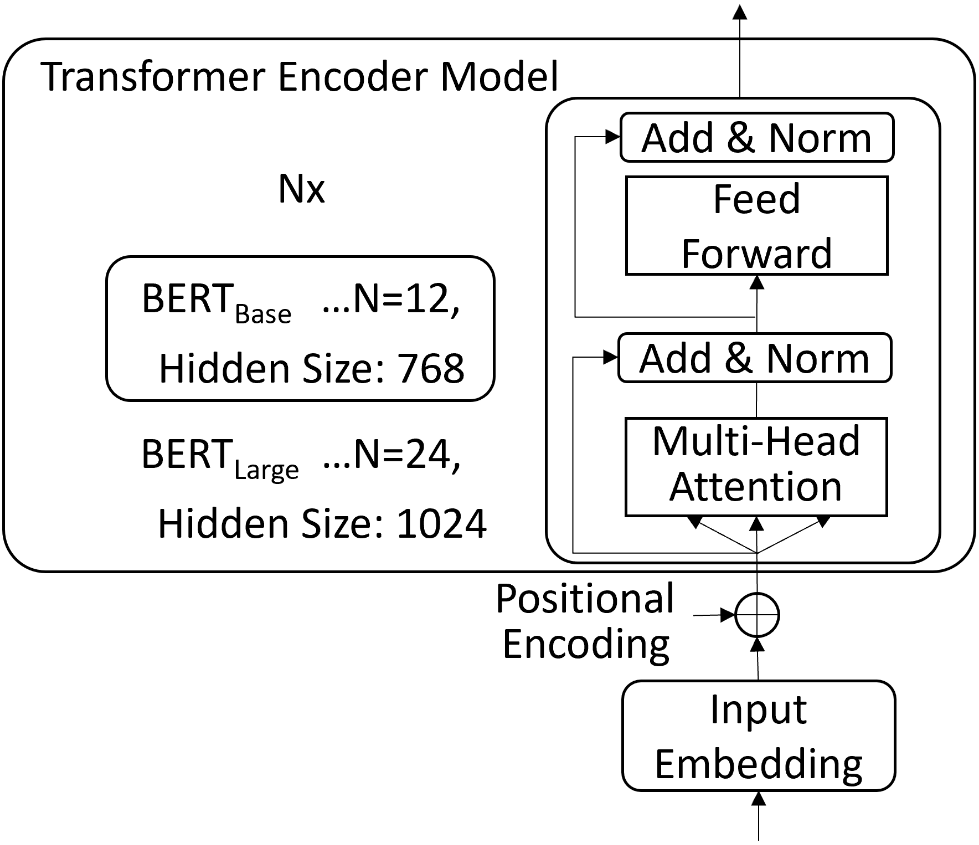
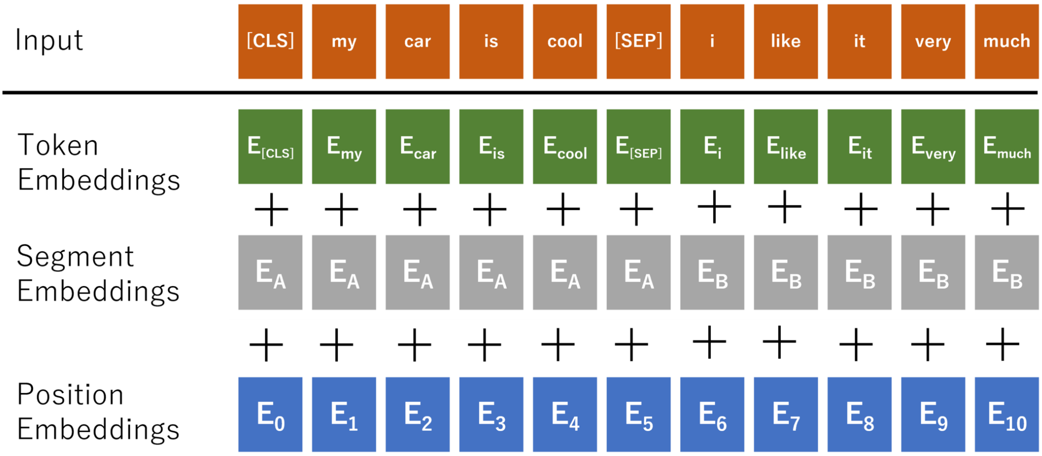


Figure 2. Embeddings of input sentence



Input	[CLS]	my	car	is	cool	[SEP]	i	like	it	very	much
Token Embeddings	$E_{[CLS]}$	E_{my}	E_{car}	E_{is}	E_{cool}	$E_{[SEP]}$	E_{i}	E_{like}	E_{it}	E_{very}	E_{much}
	+	+	+	+	+	+	+	+	+	+	+
Segment Embeddings	E_A	E_A	E_A	E_A	E_A	E_A	E_B	E_B	E_B	E_B	E_B
	+	+	+	+	+	+	+	+	+	+	+
Position Embeddings	E_0	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8	E_9	E_{10}

Table 1. Breakdown of harmful expression corpus

2-class	Harmful			Clean
4-class	**Violence**	**Hate**	**Crime**	**Other**
# of tweets	501	226	64	1,775
# of vocabulary	3,781	2,398	707	9,452

3.3 Dealing with Imbalanced Data

The corpus of harmful expressions constructed here was difficult to collect on a large scale because of its characteristics. In addition, the number of cases in the class “crime” was only 64, so it was easy to predict that the class imbalance of the data affected the accuracy. Resampling and data expansion were often used when dealing with imbalanced data in ML.

Owing to the nature of tweet data, it was difficult to create pseudo data. This was because changing, adding, or deleting words in a tweet sentence, rearranging the order of words, or changing the endings of words could significantly change the nuance. Therefore, even if the pseudo data made sense as sentences, harmful utterances might become less harmful and non-harmful utterances might become harmful. This type of data expansion introduced noise, which reduced the accuracy of the model.

Here, it is used that a resampling technique for class imbalance data. There were two types of resampling: undersampling to match the number of data in the minority class and oversampling to match the number of data in the majority class.

In oversampling, there is a method of restorative extraction that selects the same data many times (random oversampling), but it is prone to overfitting. Therefore, several techniques have been proposed to edit the data and generate new data.

The synthetic minority oversampling technique (SMOTE) (Chawla et al., 2002) is a technique that uses an algorithm called the k-nearest neighbor method to interpolate data by creating new data between one class of data and its neighboring data of the same class. Many methods have been proposed to extend SMOTE. These include SVM-SMOTE (Nguyen et al., 2009), SMOTE-ENN (Batista et al., 2004), and Borderline-SMOTE (Han et al., 2005), which perform oversampling based on boundaries. Another oversampling method called adaptive synthetic sampling (ADASYN) (He et al., 2008) considers the density of data in the feature space and generates a large number of minority class data, where there are few minority class data.

As mentioned, applying oversampling methods such as SMOTE is the same as creating the data pseudo-identically. Therefore, if high accuracy was obtained for data in the corpus, the same accuracy might not be obtained for unknown data. However, using class-imbalanced data for training caused the data to be easily classified into the majority class. Therefore, here, they are used that SMOTE and its extensions and ADASYN to deal with imbalanced data. To use oversampling methods such as SMOTE, we used the class module included in the imbalanced-learn library for Python.

3.4 Proposed Model

Here, it is used that a two-class classification of “harmful” and “clean,” and a four-class classification of “violence,” “hate,” “crime,” and “other.” In the case of multiple features, there were five input features: BERT-CLS (vector of CLS tokens), BERT-AVG (average vector of BERT variance representations), BoW-SVD (BoW vector compressed to 512 dimensions by SVD), and dependency relation vector (DRV), which is average vector of harmful representations and BERT vectors of words in the dependency relationship, and DistilBERT. CaboCha[2] was used to extract the word-word association relations. CaboCha is Japanese dependency structure analyzer based on Support Vector Machines.

For the single feature model, it is used that only one type of input feature and four types of oversampling methods: SMOTE, ADASYN, borderline SMOTE, SVM-SMOTE, and SMOTE-ENN.

Fig. 3 shows the proposed model (four-class classification). Network (a) in the figure shows the case where multiple features were used as input, and Network (b) shows the case where only one type of feature is used as input. Network (c) was a multi-task learning network that learned four-class and two-class classification models simultaneously.

In addition, by using the word vectors and the relationship between words using SVD, we believed that we could extract features flexibly even for types of sentences that did not appear in pre-trained models such as BERT. The major difference between our model and the hate-speech detection model by Kapli et al. (2020) is that the proposed model can classify two classes and four classes simultaneously, and the features we use are different from those used in their study.

Following are the network parameters for training.

- Activation function for middle layers: ReLU
- Activation function for output layers: softmax
- Number of training epochs: 10
- Optimizer: Adam
- Loss function: categorical crossentropy
- Loss weights: 0.01 (four-class), 0.02 (two-class)

3.5 MBTI and Harmful Type

The Myers-Briggs Type Indicator (MBTI) (Myers, 1995) is an index used in the field of psychology to classify personality into 16 types. Here, it is analyzed that the similarity between the tweets in the corpus of harmful expressions and the tweets randomly selected for each personality type of MBTI to determine which personality types tended to have which types of harmful expressions.

For the tweet features, it is used that Sentence BERT, which was effective in calculating sentence-to-sentence similarity. For similarity, cosine similarity is used.

The number of tweets with a cosine similarity value greater than 0.9 was counted for each personality type and for each type of harmful expression (violence, hate, crime), and the standardized count was used as the harmfulness score.

$$count_{k,l,i,j} = \begin{cases} 1 & if \quad \text{cosine}\left(V_{sb}\left(t_{ij}\right), V_{sb}\left(t_{kl}\right)\right) \geq 0.9 \\ 0 & if \quad \text{cosine}\left(V_{sb}\left(t_{ij}\right), V_{sb}\left(t_{kl}\right)\right) < 0.9 \end{cases} \tag{1}$$

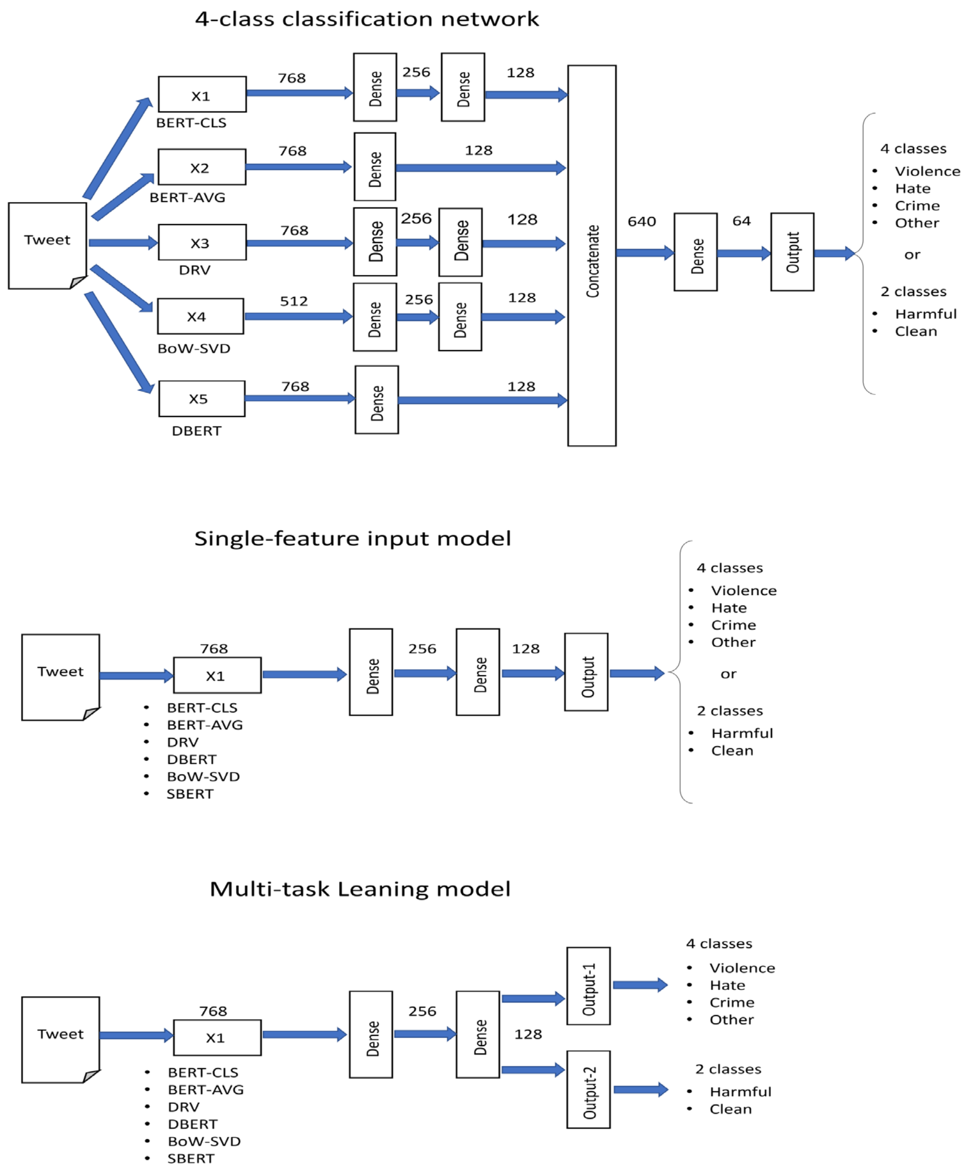
$$harm\_count_{k,p} = \sum_{l} \sum_{i} \sum_{j} count_{k,l,i,j}$$

Equation 1 shows the formula for calculating the *harm_count*. $V_{sb}(t_{ij})$ is the SentenceBERT vector of the *j*-th tweet of user *i* of personality type *p*. $V_{sb}(t_{kl})$ is the SentenceBERT vector of the *l*-th tweet of harm type *k*. $V_{sb}(t_{kl})$ is the SentenceBERT vector of the *l*-th tweet of harm type *k*. $\text{cosine}\left(V_{sb}\left(t_{ij}\right), V_{sb}\left(t_{kl}\right)\right)$ indicates the cosine similarity between $V_{sb}\left(t_{ij}\right)$ and $V_{sb}\left(t_{kl}\right)$

Also shown in Equation 2 is the formula for calculating the harm score (*harmscore*), standardized by *harm_count,* where *harm_count*$_{k,p}$ is the harm frequency of harmful type *k* in personality type *p*, and $\overline{harm\_count_{k,p}}$ is its mean value. $s_{k,p}$ is the standard deviation of the *harm_count*$_{k,p}$.

$$harmscore_{k,p} = \frac{harm\_count_{k,p} - \overline{harm\_count_{k,p}}}{s_{k,p}} \tag{2}$$

Figure 3. Architecture of neural networks for harmful tweet classification



We also analyzed how many expressions in the harmful expression list appeared in each personality type. Furthermore, it is analyzed that which type of user's tweets tended to be judged as each personality type based on the results of classification using the proposed method, a four-class classification model.

Table 2 shows an overview of the MBI corpus used in the analysis. The corpus consisted of users who tweeted the results of their diagnoses (diagnosed personality patterns) on the personality assessment site 16 Personalities[3], and the tweets were randomly extracted for each pattern.

Table 2. Breakdown of MBTI corpus

type	# of tweets	# of vocabulary	type	# of tweets	# of vocabulary
ENFJ	11,632	21,926	INFJ	11,259	22,370
ENFP	11,539	21,947	INFP	11,632	21,855
ENTJ	11,632	22,483	INTJ	11,632	22,923
ENTP	11,632	21,259	INTP	11,632	22,017
ESFJ	11,632	21,608	ISFJ	11,632	21,196
ESFP	11,632	21,295	ISFP	11,632	20,776
ESTJ	11,632	23,071	ISTJ	11,632	22,432
ESTP	11,632	23,015	ISTP	11,632	21,548

4. RESULTS AND ANALYSIS

4.1 Evaluation Method

The 10-fold cross-validation method was used to classify 2,566 sentences. The training data and test data were split at a ratio of 9:1, and the training data were further split randomly into training and validation data at a ratio of 8:2. The performance of the classification was evaluated using Recall (R), Precision (P), F1-score (F1), Accuracy, Macro Average F1-score (Macro-F1), weighted average F1-score (Weighted-F1) for four-class classification, and receiver operating characteristic (ROC) curve for two-class classification; in the case of two-class classification, a ROC curve was created, and the area under the curve (AUC) was calculated.

Equations 3 through 8 show the calculation formulas for Recall, Precision, F1-score, accuracy, Macro-F1, and Weighted-F1. TP_c was true positives; FP_c false positives; FN_c false negatives; TN_c true negatives for each category; c. C was the set of harmful classes; and c harmful class. N_c was the number of data points for each category c.

$$\text{Precision}_c = \frac{TP_c}{TP_c + FP_c} \tag{3}$$

$$\text{Recall}_c = \frac{TP_c}{TP_c + FN_c} \tag{4}$$

$$\text{F1-score}_c = 2 \cdot \frac{\text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c} \tag{5}$$

$$\text{Accuracy} = \frac{\sum_{c \in C} \left(TP_c + TN_c \right)}{\sum_{c \in C} \left(TP_c + TN_c + FP_c + FN_c \right)} \tag{6}$$

$$\text{Macro-F1} = \frac{1}{\left|C\right|}\sum_{c \in C} \text{F1-score}_{c} \tag{7}$$

$$\text{Weighted-F1} = \frac{1}{\left|C\right|}\sum_{c \in C} N_{c} \times \text{F1-score}_{c} \tag{8}$$

4.2 Result of Classification

Fig. 4 shows the results of the harmful classification (four-class classification). Fig. 5 shows the AUCs for the experimental results of the two-class classification.

These results showed that in the four-class classification, the model with multiple inputs tended to show a higher overall performance as features. In multi-task learning, the model with multiple inputs showed an accuracy of almost 70%. For two-class classification, BERT-AVG and others showed high performance, even without resampling methods. In summary, for complex classifications such as the four-class classification, either multiple features or resampling methods such as SMOTE or borderline SMOTE were effective in correcting class imbalance, and multi-task learning showed relatively higher performance. However, for a simple two-class classification of harmful or clean, multiple inputs were not very effective, and resampling and multi-task learning were not very effective.

The learning history for multitask learning is illustrated in Figure 8. The left side shows the trends of training accuracy, validation accuracy, training loss, and validation loss for the four-class classification and the right side shows that for the two-class classification. The horizontal axis represents training epochs. The results showed that the value of loss remained unchanged as epochs progressed, and the most stable model was the model with multiple inputs in multitask learning. However, the validation accuracy for all the models approached zero as the training progressed. This could be attributed to overtraining because the number of data used in this experiment was not very large.

The sentence BERT vector was visualized using uniform manifold approximation and projection (UMAP) (McInnes et al., 2018). We used umap-learn[4] as a library for UMAP. Random under-sampling was used to adjust the class balance. Fig. 9 shows the visualization results. From this result, it can be seen that “crime” is easily separated from other categories. The fact that “other” was widely distributed, and also suggested that there were few common points in the “other” class.

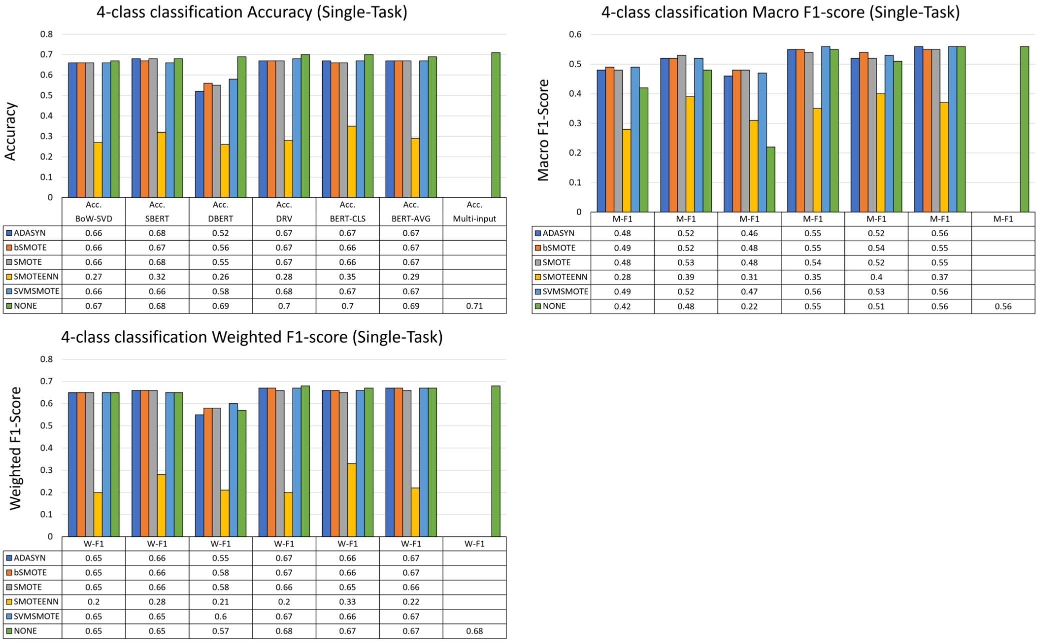
Here, in the single-task model, the confusion matrix of BERT-CLS without resampling/SMOTE, Sentence BERT without resampling/SMOTE, Single-Task with Mutli-input, and Multi-Task was visualized by a heatmap, (see Fig. 10). “Hate” and “violence” tended to be misclassified as “other” in a high percentage of cases. The percentage of misclassification of “crime” was lower than that of the other harmful classes, which was expected from the distribution tendency of the feature values.

4.3 Relationship between Harmful Category and MBTI

In this section, we present the results of calculating the harmful scores based on Equation (2) and the results of classifying tweets using the harmful classification model that performed best in the experiments described in the previous section to obtain the classification tendency for each personality type of users’ MBTI.

Fig. 11 shows the result of the harmful scores (*harmscore*) is calculated. It can be observed that the following:

Figure 4. Accuracy, Macro-F1 and Weighted-F1 (single-task)

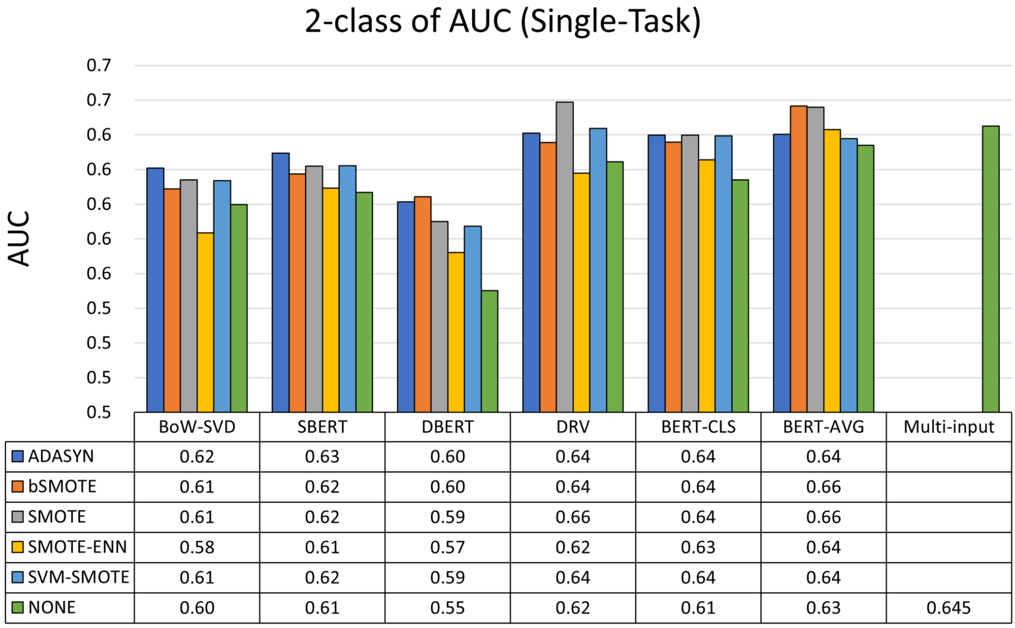


	Acc. BoW-SVD	Acc. SBERT	Acc. DBERT	Acc. DRV	Acc. BERT-CLS	Acc. BERT-AVG	Acc. Multi-input
🟦 ADASYN	0.66	0.68	0.52	0.67	0.67	0.67	
🟧 bSMOTE	0.66	0.67	0.56	0.67	0.66	0.67	
⬜ SMOTE	0.66	0.68	0.55	0.67	0.66	0.67	
🟨 SMOTEENN	0.27	0.32	0.26	0.28	0.35	0.29	
🟦 SVMSMOTE	0.66	0.66	0.58	0.68	0.67	0.67	
🟩 NONE	0.67	0.68	0.69	0.7	0.7	0.69	0.71

	M-F1	M-F1	M-F1	M-F1	M-F1	M-F1	M-F1
🟦 ADASYN	0.48	0.52	0.46	0.55	0.52	0.56	
🟧 bSMOTE	0.49	0.52	0.48	0.55	0.54	0.55	
⬜ SMOTE	0.48	0.53	0.48	0.54	0.52	0.55	
🟨 SMOTEENN	0.28	0.39	0.31	0.35	0.4	0.37	
🟦 SVMSMOTE	0.49	0.52	0.47	0.56	0.53	0.56	
🟩 NONE	0.42	0.48	0.22	0.55	0.51	0.56	0.56

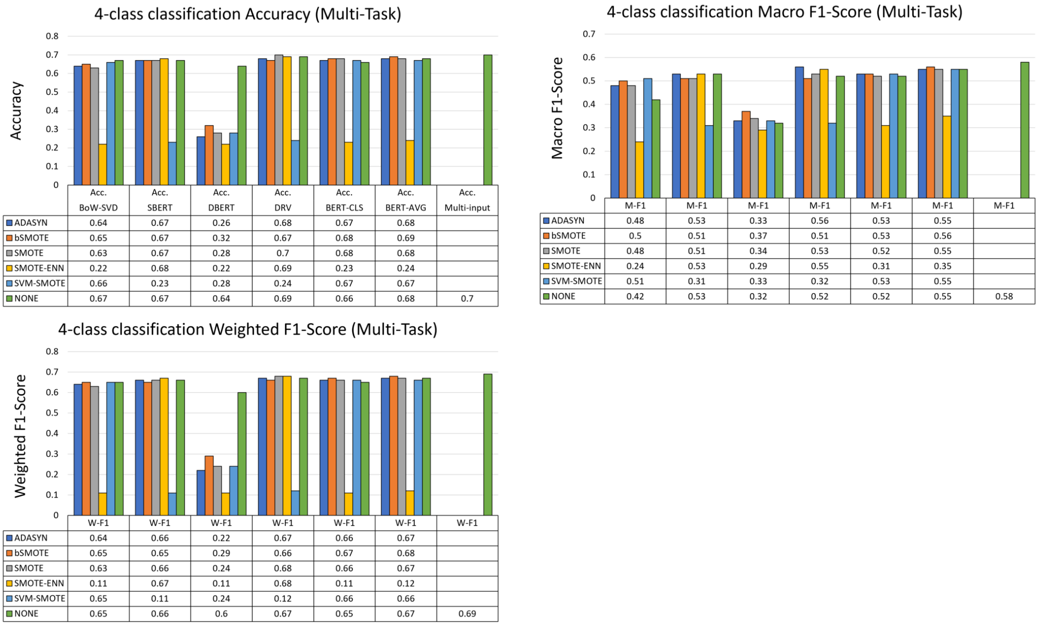
	W-F1	W-F1	W-F1	W-F1	W-F1	W-F1	W-F1
🟦 ADASYN	0.65	0.66	0.55	0.67	0.66	0.67	
🟧 bSMOTE	0.65	0.66	0.58	0.67	0.66	0.67	
⬜ SMOTE	0.65	0.66	0.58	0.66	0.65	0.66	
🟨 SMOTEENN	0.2	0.28	0.21	0.2	0.33	0.22	
🟦 SVMSMOTE	0.65	0.65	0.6	0.67	0.66	0.67	
🟩 NONE	0.65	0.65	0.57	0.68	0.67	0.67	0.68

Figure 5. Comparison of AUC (single-task)



	BoW-SVD	SBERT	DBERT	DRV	BERT-CLS	BERT-AVG	Multi-input
🟦 ADASYN	0.62	0.63	0.60	0.64	0.64	0.64	
🟧 bSMOTE	0.61	0.62	0.60	0.64	0.64	0.66	
⬜ SMOTE	0.61	0.62	0.59	0.66	0.64	0.66	
🟨 SMOTE-ENN	0.58	0.61	0.57	0.62	0.63	0.64	
🟦 SVM-SMOTE	0.61	0.62	0.59	0.64	0.64	0.64	
🟩 NONE	0.60	0.61	0.55	0.62	0.61	0.63	0.645

Figure 6. Accuracy, macro-F1 and weighted-F1 (multi-task)



4-class classification Accuracy (Multi-Task)

	Acc.	Acc.	Acc.	Acc.	Acc.	Acc.	Acc.
	BoW-SVD	SBERT	DBERT	DRV	BERT-CLS	BERT-AVG	Multi-input
🟦 ADASYN	0.64	0.67	0.26	0.68	0.67	0.68	
🟧 bSMOTE	0.65	0.67	0.32	0.67	0.68	0.69	
⬜ SMOTE	0.63	0.67	0.28	0.7	0.68	0.68	
🟨 SMOTE-ENN	0.22	0.68	0.22	0.69	0.23	0.24	
🟦 SVM-SMOTE	0.66	0.23	0.28	0.24	0.67	0.67	
🟩 NONE	0.67	0.67	0.64	0.69	0.66	0.68	0.7

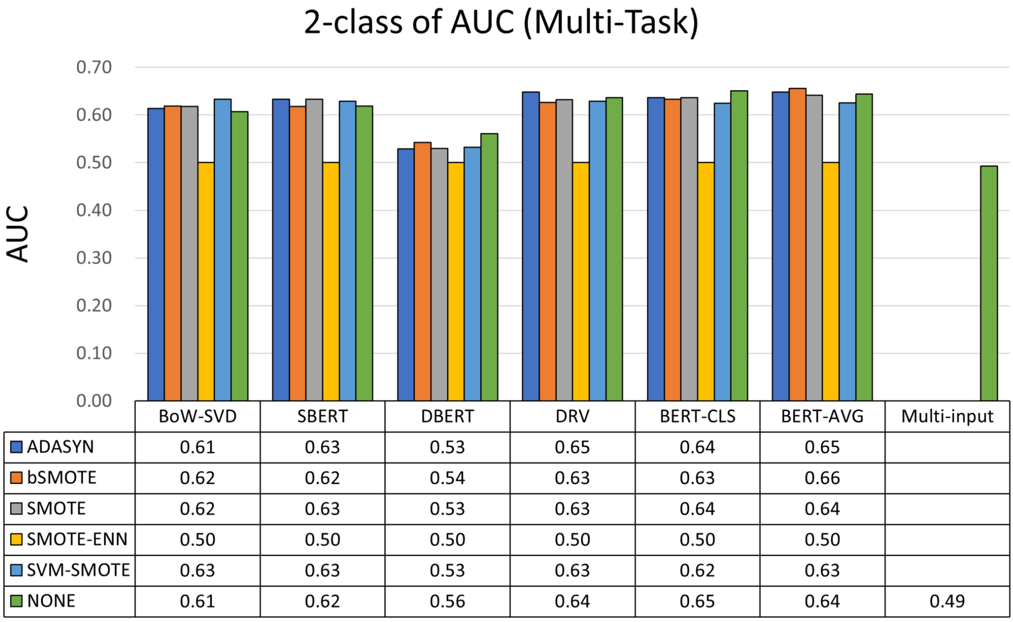
4-class classification Macro F1-Score (Multi-Task)

	M-F1	M-F1	M-F1	M-F1	M-F1	M-F1	M-F1
🟦 ADASYN	0.48	0.53	0.33	0.56	0.53	0.55	
🟧 bSMOTE	0.5	0.51	0.37	0.51	0.53	0.56	
⬜ SMOTE	0.48	0.51	0.34	0.53	0.52	0.55	
🟨 SMOTE-ENN	0.24	0.53	0.29	0.55	0.31	0.35	
🟦 SVM-SMOTE	0.51	0.31	0.33	0.32	0.53	0.55	
🟩 NONE	0.42	0.53	0.32	0.52	0.52	0.55	0.58

4-class classification Weighted F1-Score (Multi-Task)

	W-F1	W-F1	W-F1	W-F1	W-F1	W-F1	W-F1
🟦 ADASYN	0.64	0.66	0.22	0.67	0.66	0.67	
🟧 bSMOTE	0.65	0.65	0.29	0.66	0.67	0.68	
⬜ SMOTE	0.63	0.66	0.24	0.68	0.66	0.67	
🟨 SMOTE-ENN	0.11	0.67	0.11	0.68	0.11	0.12	
🟦 SVM-SMOTE	0.65	0.11	0.24	0.12	0.66	0.66	
🟩 NONE	0.65	0.66	0.6	0.67	0.65	0.67	0.69

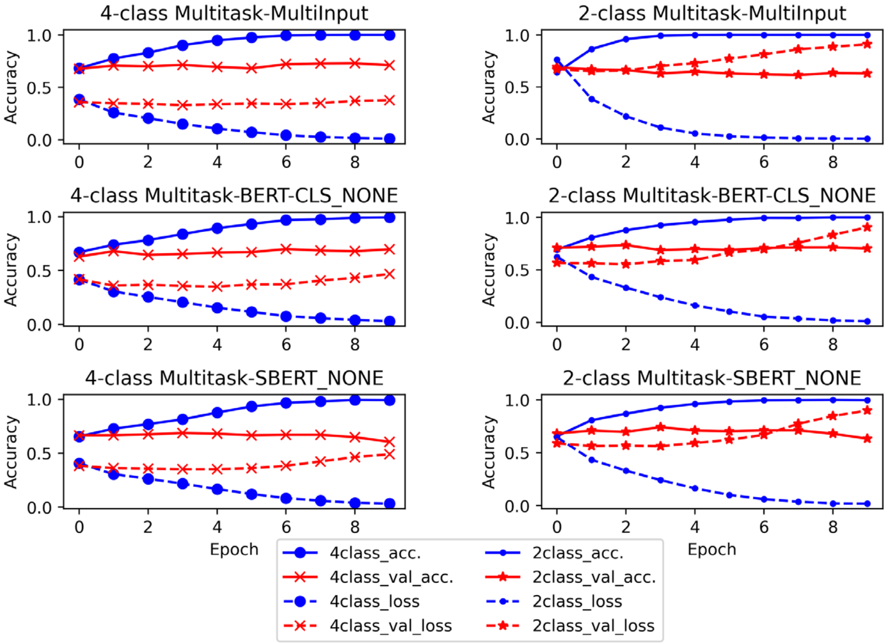
Figure 7. Comparison of AUC (multi-task)



2-class of AUC (Multi-Task)

	BoW-SVD	SBERT	DBERT	DRV	BERT-CLS	BERT-AVG	Multi-input
🟦 ADASYN	0.61	0.63	0.53	0.65	0.64	0.65	
🟧 bSMOTE	0.62	0.62	0.54	0.63	0.63	0.66	
⬜ SMOTE	0.62	0.63	0.53	0.63	0.64	0.64	
🟨 SMOTE-ENN	0.50	0.50	0.50	0.50	0.50	0.50	
🟦 SVM-SMOTE	0.63	0.63	0.53	0.63	0.62	0.63	
🟩 NONE	0.61	0.62	0.56	0.64	0.65	0.64	0.49

Figure 8. History of multi-task learning



- Most extroverts (E) tended to have high adverse scores.
- Introverts (I) tended to have low harmful scores, except for INTJ and INTP.
- Intuitive (N) and thoughtful (T) introverts tended to have extremely high harmful scores.
- The IE, NS, and TF indices affected the change in scores.
- Judgmental (J) or perceptual (P) measures had little effect on the harmful score.

Furthermore, the presence or absence of harmful expressions was analyzed for each personality type of the MBTI. Table 3 presents the results.

Next, it is analyzed that the classification results using the single-input feature harmful prediction model (SMOTE, feature type: BERT-CLS) trained in the experiments described in the previous section, and Fig. 12 shows the results. The vertical axis in the graph represents the standardized value of the number of times the classification was made.

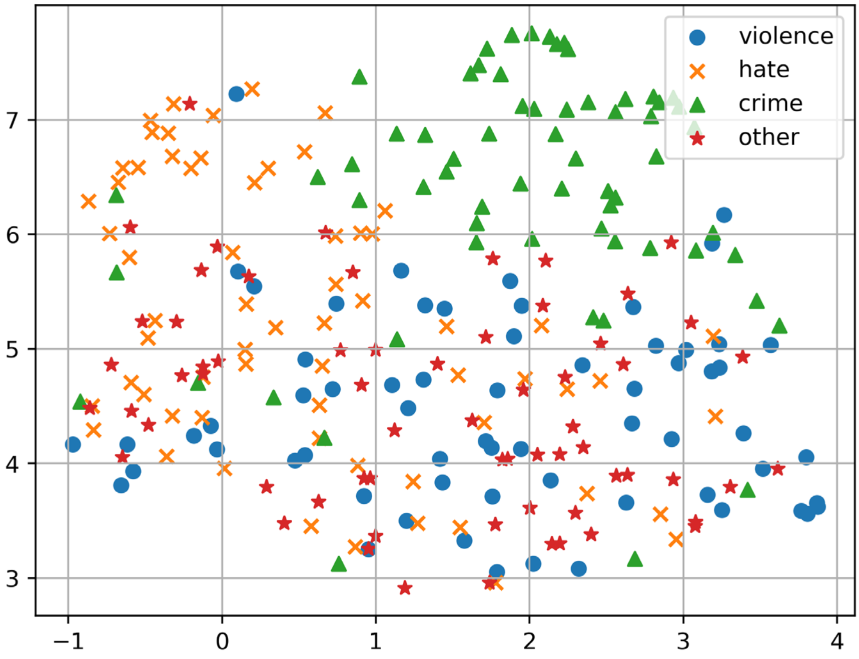
INFJ was the personality type that contained the least number of harmful expressions. However, ENTP was the personality type that contained the most harmful expressions, and the classification results showed high values for all three types: violence, hate, and crime.

In addition, the presence or absence of harmful expression was likely to be a cue for classification. However, as in the case of the ISTP, there were cases where the values of violence and crime were high even though harmful expressions were not used so much, so there was a possibility that harmful statements were made using unknown expressions not included in the dictionary of harmful expressions.

In using the BERT features, it is detected that unknown expressions and harmfulness that could be interpreted from the whole sentence.

In this study, we investigated personality patterns with similar tendencies based on the results of the detected models. The procedure of this investigation is as follows:

Figure 9. Visualizing of Sentence BERT embeddings

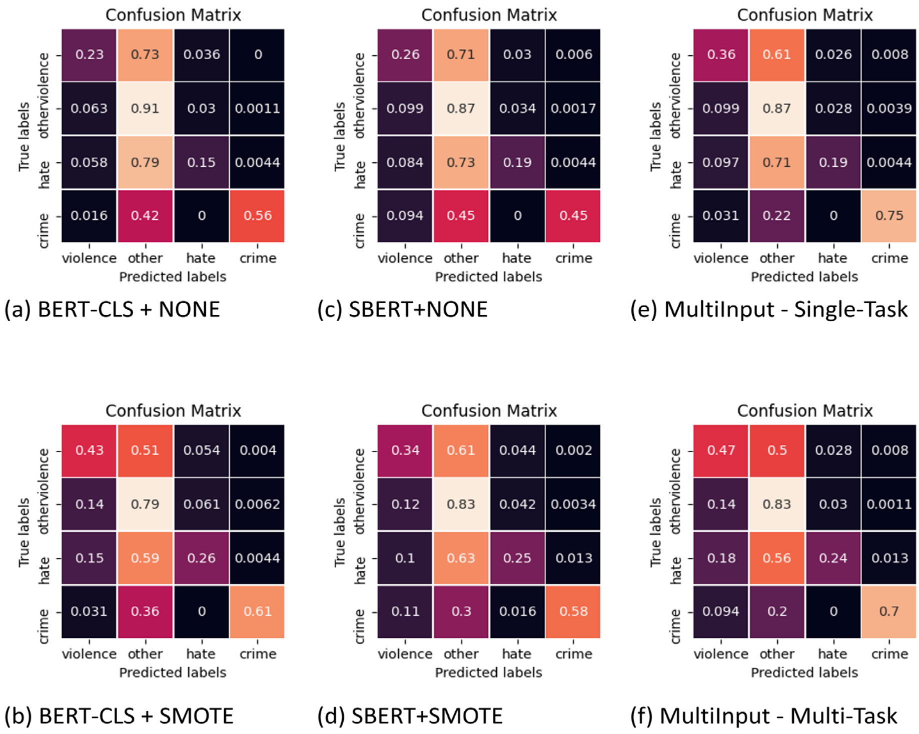


1. For each personality pattern, a vector with the distribution of the harmful score as a feature was created.
2. Grouped similar patterns by unsupervised clustering using k-means.
3. The results visualized by UMAP were compared to that of clustering.
4. The commonalities among similar personality patterns were considered.

In this study, we set the value of k (number of clusters) to 5. Figure 13 shows the visualization results. The colors of the points in the plot represent personality patterns, and the shapes of the markers represent the differences in the clusters.

There are certain areas where the same personality is distributed unevenly near the center of the figure, but they are distributed widely, and there are no clusters that are biased toward a specific personality. This indicated that there were no significant characteristics in the distribution trends of personality and harmful scores. However, a certain number of users made many harmful comments. Based on the classification of personality patterns from the content of tweets, we believe that it is possible to find users with personality patterns similar to that of those users and prevent flaming.

Figure 10. Confusion matrix



(a) BERT-CLS + NONE

(c) SBERT+NONE

(e) MultiInput - Single-Task

(b) BERT-CLS + SMOTE

(d) SBERT+SMOTE

(f) MultiInput - Multi-Task

Figure 11. Relation between MBTI type and harmful category

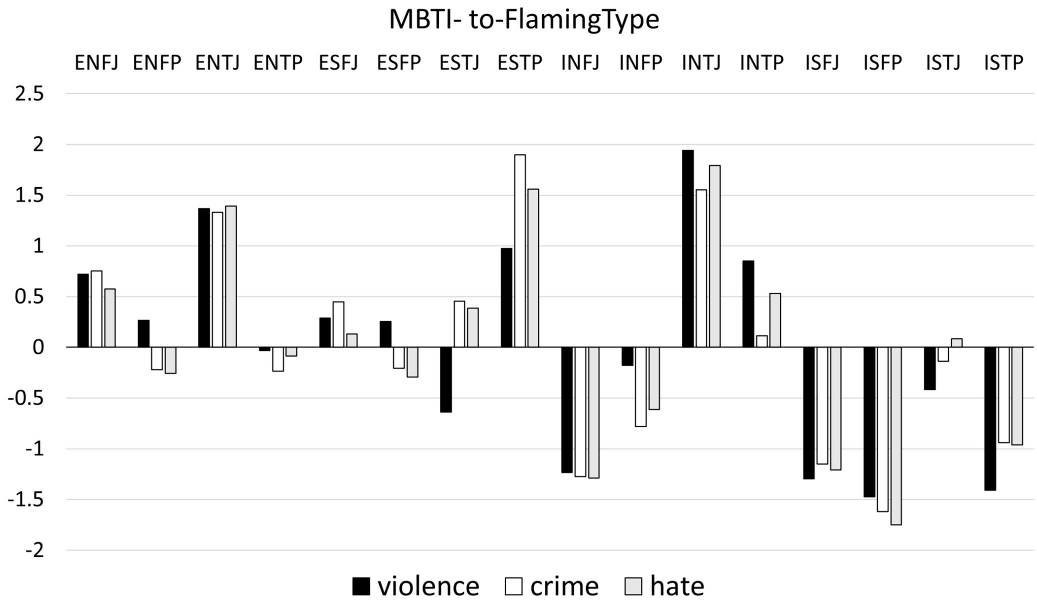


Table 3. Number of harmful expressions appeared in each MBTI

MBTI type	Num. of harmful expressions	MBTI type	Num. of harmful expressions
ENTP	610	INFP	415
ESTP	495	ESFP	375
ISTJ	470	ESFJ	370
INTP	464	ISTP	370
ENTJ	453	ENFP	369
INTJ	444	ISFP	362
ESTJ	437	ISFJ	358
ENFJ	433	INFJ	331

Figure 12. Classification result of MBTI corpus by harmful prediction model

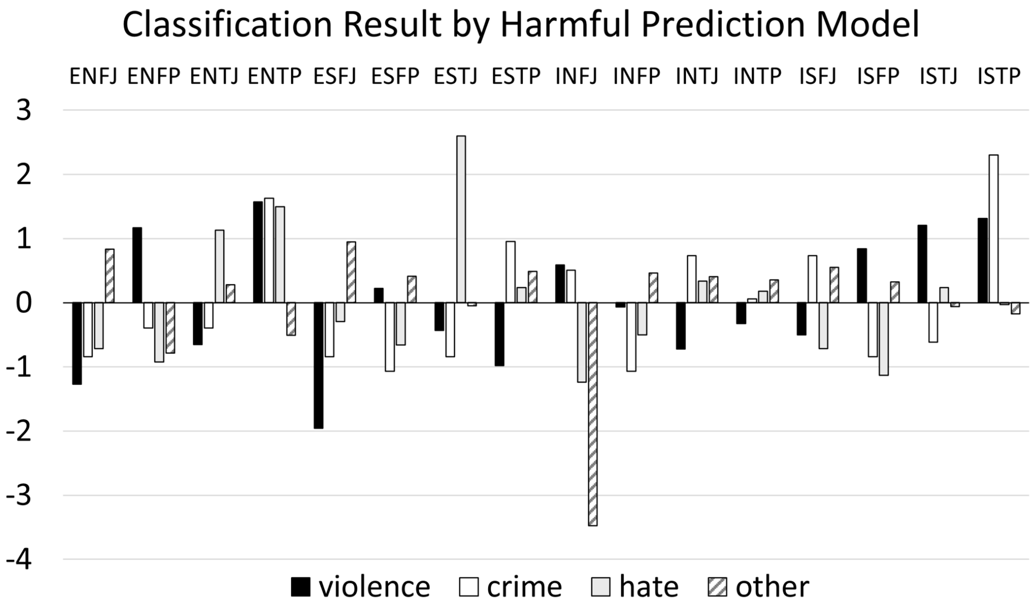
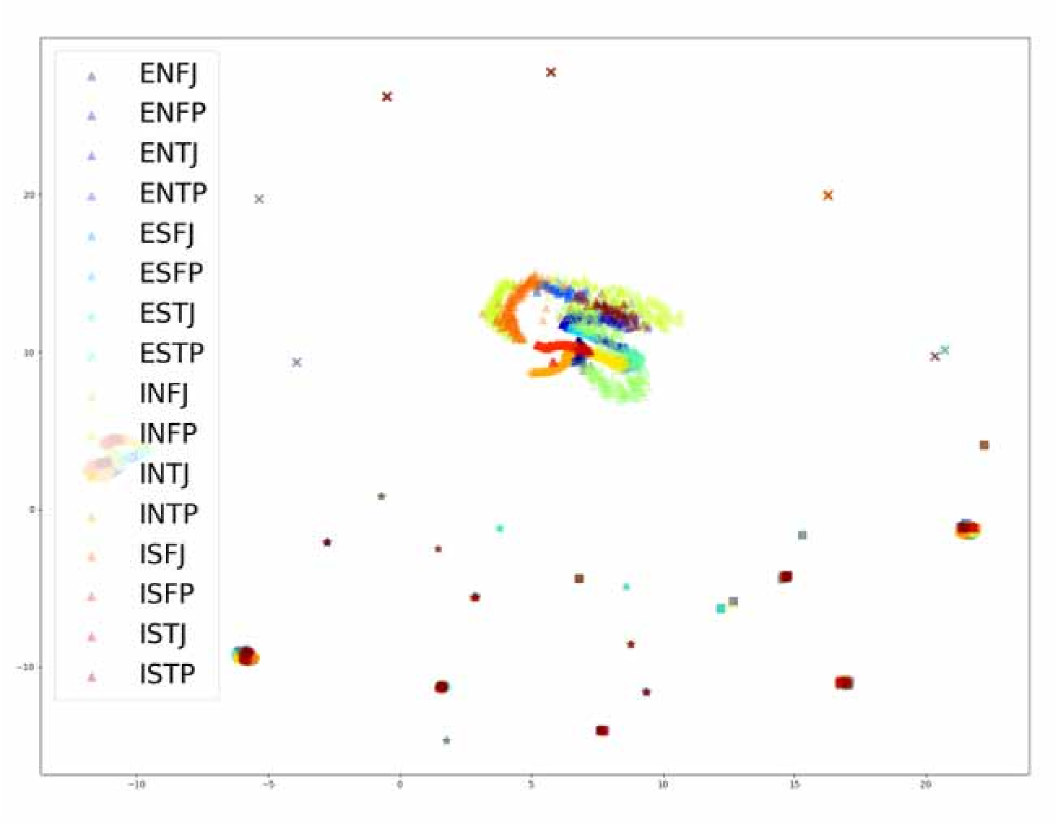


Figure 13. Visualizing of harmful score vector for each personality type



5. CONCLUSION

This paper proposed a method to analyze remarks containing harmful expressions and identify the harmful class as a method to prevent Internet flaming caused by inappropriate remarks on social media. The proposed method collected statements containing harmful expressions from Twitter and constructed a corpus of harmful statements labeled with harmful categories, such as abusive language, discriminatory remarks, and criminal threats. For this corpus, BERT features were extracted for each utterance after denoising, and a classification model for the harmful category was learned using a neural network. The problem of label imbalance was addressed using an oversampling technique. The experiment confirmed that approximately 70% of the data could be classified by accuracy, although there was some bias in the data. In the future, we would like to investigate the accuracy of the accuracy of harmful or non-harmful statements that do not contain harmful expressions.

We also analyzed Twitter users' tweet data associated with the MBTI personality test using the harmful class classification model and the harmful expression dictionary. The results showed that there was a limited relationship between certain personality patterns, harmful categories, and harmful expressions. This suggested that if the personality tendencies of users who post harmful tweets were known in advance, it would be possible to prevent Internet flaming and actions that contribute to flaming by making harmful judgments based on the text of tweets in advance.

Kishima et al. (2021) proposed a method for automatically analyzing the MBTI from the content of tweets. The proposed method can predict the personality of the target user with 70% accuracy if the user's tweets are obtained in advance; therefore, it is expected to contribute to rapid flame detection by combining it with the proposed method.

In future research, we will clarify which combinations of personality patterns and psychological patterns lead to harmful comments, which will be useful in constructing a monitoring system for preventing flaming that can be applied not only to Twitter but also to social media in general. Furthermore, it is believed that these findings will be useful when introducing personality patterns into AI chatbots.

ACKNOWLEDGMENT

This work was supported by the 2021 SCAT Research Grant and JSPS KAKENHI Grant Number JP20K12027, JP21K12141.

REFERENCES

Amirhosseini, M., & Kazemian, H. (2020). Machine Learning Approach to Personality Type Prediction Based on the Myers–Briggs Type Indicator®. *Multimodal Technologies and Interaction*, *4*(1), 9. doi:10.3390/mti4010009

Babakov, N., Logacheva, V., Kozlova, O., Semenov, N., & Panchenko, A. (2021). Detecting inappropriate messages on sensitive topics that could harm a company's reputation. *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*, 26-36.

Ball-Burack, A., Lee, M. S., Cobbe, J., & Singh, J. (2021). Differential tweetment: mitigating racial dialect bias in harmful tweet detection. *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 116-128. doi:10.1145/3442188.3445875

Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explorations*, *6*(1), 20–29. doi:10.1145/1007730.1007735

Bozyiğit, A., Utku, S., & Nasiboğlu, E. (2019). Cyberbullying detection by using artificial neural network models. *Proceedings of the 4th International Conference on Computer Science and Engineering (UBMK).* doi:10.1109/UBMK.2019.8907118

Bozyiğit, A., Utku, S., & Nasibov, E. (2021). Cyberbullying detection: Utilizing social media features. *Expert Systems with Applications*, *179*(1), 115001. doi:10.1016/j.eswa.2021.115001

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. doi:10.1613/jair.953

Choi, Y.-J., Jeon, B.-J., & Kim, H.-W. (2021). Identification of key cyberbullies: A text mining and social network analysis approach. *Telematics and Informatics*, *56*, 101504. doi:10.1016/j.tele.2020.101504

Choong, E. J., & Varahan, K. D. (2021). Predicting judging-perceiving of Myers-Briggs Type Indicator (MBTI) in online social forum. *PeerJ*, *9*, e11382. doi:10.7717/peerj.11382 PMID:34221705

Çürük, E., Acı, Ç., & Eşsiz, E. S. (2018). The effects of attribute selection in artificial neural network based classifiers on cyberbullying detection. *Proceedings of the 3rd International Conference on Computer Science and Engineering (UBMK).*

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT*, *2019*, 4171–4186.

Ezpeleta, E., Garitano, I., Zurutuza, U., & Hidalgo, M. (2017). Short Messages Spam Filtering Combining Personality Recognition and Sentiment Analysis. *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, *25*(2), 175–189. doi:10.1142/S0218488517400177

Fortunatus, M., Anthony, P., & Charters, S. (2020). Combining textual features to detect cyberbullying in social media posts. *Procedia Computer Science*, *176*, 612–621. doi:10.1016/j.procs.2020.08.063

Fuchs, D. (2018). The dangers of human-like bias in machine-learning algorithms. *Missouri S&T's Peer to Peer, 2*(1), 1-14.

Gehring, J., Auli, M., Grangier, D., & Dauphin, Y. (2017). A convolutional encoder model for neural machine translation. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, *1*, 123-135. doi:10.18653/v1/P17-1012

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471. doi:10.1162/089976600300015015 PMID:11032042

Han, H., Wang, W. Y., & Mao, B. H. (2005). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. *Proceedings of the International Conference on Intelligent Computing, ICIC 2005*, 878-887.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: adaptive synthetic sampling approach for imbalanced learning. *Proceedings of 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence).*

Ishisaka, T., & Yamamoto, K. (2010). Extraction of swear words from 2channel. *Proceedings of the 16th annual meeting of the Association for Natural Language Processing*, 178-181.

Jacobs, A. (2019). Sentiment Analysis for Words and Fiction Characters From the Perspective of Computational (Neuro-)Poetics. *Frontiers in Robotics and AI*, *6*(53), 53. doi:10.3389/frobt.2019.00053 PMID:33501068

Kapli, P., & Ekbal, A. (2020). A deep neural network based multi-task learning approach to hate speech detection. *Knowledge-Based Systems*, *210*, 106458. doi:10.1016/j.knosys.2020.106458

Karayiğit, H., Aci, C., & Akdagli, A. (2021). Detecting abusive Instagram comments in Turkish using convolutional neural network and machine learning methods. *Expert Systems with Applications*, *17415*(114802), 114802. doi:10.1016/j.eswa.2021.114802

Keh, S. S., & Cheng, I.-T. (2019). *Myers-Briggs Personality Classification and Personality-Specific Language Generation Using Pre-trained Language Models.* arXiv preprint arXiv:1907.06333

Khan, A. S., Ahmad, H., Zubair, M., & Asghar, Z. (2020). Personality Classification from Online Text using Machine Learning Approach. *International Journal of Advanced Computer Science and Applications*, *11*(3). Advance online publication. doi:10.14569/IJACSA.2020.0110358

Kishima, R., Matsumoto, K., Yoshida, M., & Kita, K. (2021). Construction of MBTI personality estimation model considering emotional information. *Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation.*

Lingam, R. A., & Aripin, N. (2017). Comments on fire! classifying flaming comments on YouTube videos in Malaysia. *Jurnal Komunikasi: Malaysian Journal of Communication*, *33*(4), 104–118.

López-Vizcaíno, M., Nóvoa, F. J., Carneiro, V., & Cacheda, F. (2021). Early detection of cyberbullying on social media networks. *Future Generation Computer Systems*, *118*, 219–229. doi:10.1016/j.future.2021.01.006

Matsumoto, K., Tsuchiya, S., Miyake, T., Yoshida, M., & Kita, K. (2018). Flame prediction based on harmful expression judgement using distributed representation. *International Journal of Technology and Engineering Studies*, *4*(1), 7–15.

McInnes, L., Healy, J., & Melville, J. (2018). UMAP: Uniform manifold approximation and projection for dimension reduction. *Journal of Open Source Software*, *3*(29), 861.

Mushtaq, Z., Ashraf, S., & Sabahat, N. (2020). Predicting MBTI Personality type with K-means Clustering and Gradient Boosting. *Proceedings of IEEE 23rd International Multitopic Conference (INMIC)*. doi:10.1109/INMIC50486.2020.9318078

Myers, I. B., & Myers, P. B. (1995). *Gifts differing: understanding personality type*. Davies-Black Publishing.

Nguyen, H. M., Cooper, E. W., & Kamei, K. (2009). Borderline over-sampling for imbalanced data classification. *International Journal of Knowledge Engineering and Soft Data Paradigms*, *3*(1), 4–21. doi:10.1504/IJKESDP.2011.039875

Omar, A., Mahmoud, T. M., & Abd-El-Hafeez, T. (2020). Comparative performance of machine learning and deep learning algorithms for Arabic hate speech detection in OSNs. *Proceedings of the 1st International Conference on Artificial Intelligence and Computer Visions, AICV 2020*, 247-257. doi:10.1007/978-3-030-44289-7_24

Ozawa, S., Yoshida, S., Kitazono, J., Sugawara, T., & Haga, T. (2016). A sentiment polarity prediction model using transfer learning and its application to SNS flaming event detection. *Proceedings of IEEE Symposium Series on Computational Intelligence (SSCI 2016)*. doi:10.1109/SSCI.2016.7849868

Perera, A., & Fernando, P. (2021). Accurate Cyberbullying Detection and Prevention on Social Media. *Procedia Computer Science*, *181*, 605–611. doi:10.1016/j.procs.2021.01.207

Ptaszynski, M., Masui, F., Nakajima, Y., Kimura, Y., Rzepka, R., & Araki, K. (2017). A method for detecting harmful entries on informal school websites using morphosemantic patterns. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, *21*(7), 1189–1201. doi:10.20965/jaciii.2017.p1189

Reimers, N., & Gurevych, I. (2019). Sentence- BERT: sentence embeddings using Siamese BERT networks. *Proceedings of 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 3982-3992. doi:10.18653/v1/D19-1410

Salsabila, G., & Setiawan, E. (2021). Semantic Approach for Big Five Personality Prediction on Twitter. *Jurnal RESTI*, *5*(4), 680–687. doi:10.29207/resti.v5i4.3197

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *Proceedings of Thirty-third Conference on Neural Information Processing Systems (NIPS2019).*

Sani, N., Isa, M., & Ashianti, L. (2017). Cyberbullying classification using text mining. *Proceedings of the 1st International Conference on Informatics and Computational Sciences (ICICoS).*

Sarna, G., & Bhatia, M. P. S. (2020). Structure-based analysis of different categories of cyberbullying in dynamic social network. *International Journal of Information Security and Privacy*, *14*(3), 1–17. doi:10.4018/IJISP.2020070101

Steinberger, J., Brychcin, T., Hercig, T., & Krejzl, P. (2017). Cross-lingual flames detection in news discussions. *Proceedings of International Conference Recent Advances in Natural Language Processing*.

Tanaka, T., & Yamaguchi, S. (2016). *Study on Interet Flaming "Who will stir things up and how will they be handled?* Keiso Publishing.

Watanabe, H., Bouazizi, M., & Ohtsuki, T. (2018). Hate speech on Twitter: A pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. *IEEE Access: Practical Innovations, Open Solutions*, *6*, 13825–13835. doi:10.1109/ACCESS.2018.2806394

Yamada, K., Sasano, R., & Takeda, K. (2019). Incorporating Textual Information on User Behavior for Personality Prediction. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, 177-182. doi:10.18653/v1/P19-2024

Yoshida, S., Kitazono, J., Ozawa, S., Sugawara, T., & Haga, T. (2016). Improving the accuracy of sentiment analysis of SNS comments using transfer learning and its application to flaming detection. *IEEJ Transactions on Electronics*, *Information Systems*, *136*(3), 340–347.

Yoshida, S., Kitazono, J., Ozawa, S., Sugawara, T., Haga, T., & Nakamura, S. (2014). Sentiment analysis for various SNS media using Naïve Bayes classifier and its application to flaming detection. *Proceedings of 2014 IEEE Symposium Series on Computational Intelligence - CIBD 2014: 2014 IEEE Symposium on Computational Intelligence in Big Data*.

Zhao, C., Wang, S., Li, D., Liu, X., Yang, X., & Liu, J. (2021). Cross-domain sentiment classification via parameter transferring and attention sharing mechanism. *Information Sciences*, *578*, 281–296.

ENDNOTES

[1] https://taku910.github.io/mecab/
[2] https://taku910.github.io/cabocha/
[3] https://www.16personalities.com/ja
[4] https://umap-learn.readthedocs.io/en/latest/