IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 26, 2021, accepted October 17, 2021, date of publication October 19, 2021, date of current version October 28, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3121508

A YouTube Spam Comments Detection Scheme Using Cascaded Ensemble Machine Learning Model

HAYOUNG OH
College of Computing and Informatics, Sungkyunkwan University, Seoul 03063, South Korea
e-mail: hyoh79@skku.edu

This work was supported by the Ministry of Education and National Research Foundation of Korea through the ''Convergence and Open Sharing System'' Project.

ABSTRACT This paper proposes a technique to detect spam comments on YouTube, which have recently seen tremendous growth. YouTube is running its own spam blocking system but continues to fail to block them properly. Therefore, we examined related studies on YouTube spam comment screening and conducted classification experiments with six different machine learning techniques (Decision tree, Logistic regression, Bernoulli Naïve Bayes, Random Forest, Support vector machine with linear kernel, Support vector machine with Gaussian kernel) and two ensemble models (Ensemble with hard voting, Ensemble with soft voting) combining these techniques in the comment data from popular music videos - Psy, Katy Perry, LMFAO, Eminem and Shakira.

INDEX TERMS Classification, data analysis, ensemble machine learning, spam comment, YouTube comment.

I. INTRODUCTION

YouTube, the world's largest video sharing site, was founded in 2005 and acquired by Google in 2006. YouTube has grown tremendously as a video content platform, with the recent shift in online content to video. At present, more than 400 hours of video are uploaded and 4.5 million videos are watched every minute on YouTube [1]. It is easy for users to watch and upload videos without any restrictions. This great accessibility has increased the number of personal media, and some of them have become online influencers.

YouTube creators can monetize if they have more than 1,000 subscribers and 4,000 hours of watch time for the last 12 months [2]. Accordingly, spam comments are being created to promote their channels or videos in popular videos. Some creators closed the comment function due to aggression such as political comments, abusive speech, or derogatory comments not related to their videos.

YouTube has its own spam filtering system, though there are still spam comments that are not being caught. In this paper, we review related studies on YouTube spam comments and propose the Cascaded Ensemble Machine Learning Model aware YouTube Spam Comments Detection Scheme to improve the performance of the model. In previous studies, various machine learning techniques were applied to each dataset to detect spam comments and compare their performance. Therefore, in this paper, we propose an ensemble machine learning method that combines the results of several models to produce the final result.

This paper is organized as follows: In Section 2, we review related work. Section 3 describes the system model and proposed techniques, and Section 4 describes the experiments and results. Then we conclude in Section 5.

II. RESEARCH METHODS AND MATERIAL

Research on detecting spam content and users focus on various fields. Many studies focused on spam on websites (e.g., portal sites and blogs). As YouTube gains popularity as a video sharing platform, spammers target it with low quality content or promotions. Since spammers that harm the YouTube community are increasing, detecting them becomes an interesting source to research. So, we divide the literature of detecting spam into two sections, spam on websites and spam on YouTube.

The associate editor coordinating the review of this manuscript and approving it for publication was Tony Thomas.

A. DETECTING SPAM ON WEBSITES

To detect untruthful reviews of the specific product on the Internet, [3] used an n-gram language model. They focused on untruthful reviews that could be duplicated from different ids. Each review was compared with all the others to identify duplicate reviews. N-gram was used to estimate word sequence like which words would be next. In other words, n-grams decompose sentences automatically, breaking them into several small pieces.

For spammers in blog comments, researchers collected 50 posts from some blogs with 1,024 comments [4]. Then the comments were manually classified as legitimate and spam comments where 32% were legitimate and 68% were spam. Since blog posts, comments, and external links in comments were written in different styles, they used different language models. Through the language modeling approach, they applied the model to the text used in the blog posts, as well as the comments and links on the posts. The similarity of each model was compared by KL-divergence, calculating the difference between the probability distributions of spam and normal data.

A language model is statistical word sequences which represent a probability distribution of the next words based on a context or previous words [5]. Reviews and websites, especially blog comments, convey the meaning of the content in context. This means that semantic analysis can be applied, so language models are used to detect spam in text data.

B. DETECTING SPAM ON YouTube

With YouTube comments, applying the same method (i.e., language modeling) doesn't work as the features of the data are different. Features of YouTube comments represent less textual descriptions and information. They are not closely relevant to the video content. So, a different approach needs to be used to find spam on YouTube. There are some studies that follow classification algorithms to detect spam videos or a set of comments and classifies them as legitimate or spam.

References [6] presents a characterization of the social or anti-social behavior of users and video attributes that can be used to distinguish spammers from legitimate users in YouTube. They first collected users randomly who uploaded at the specific time to find who and which video had a responsive connection. Test collection consisted of 592 YouTube users who were classified manually as legitimate users and spammers, creating the standards of spam types. If a responsive video advertises a specific product or contains pornographic subject regardless of the subject of the video, the user is classified as a spammer. The number of videos that these users responded to was 16,611 to 8,710 videos. After applying the classification method, SVM in this study, Spammers could be detected based on the attributes of user, video, and social network. To evaluate the results from SVM, spam metrics (TP, TN, FP, FN), accuracy, and F-measure were used. SVM is used to classify data by continuously learning, calculating, and updating how likely the input data is classified as spam. They found that YouTube users' videos as well as responded videos form social networks in YouTube.

Khan *et al.* [7] found 500 users who uploaded videos during the crawling period and collected 30,621 videos from them. 16 channel features were extracted and then used to follow an Edge Rank algorithm which functions as a recommendation system on Facebook. These features include channel age, channel average upload, view rate based on channel age, like rate based on total views, etc. Then, nine algorithms were applied to evaluate the feature set and the Bayes Network and Naive Bayesian with a Bayes classifier show 98% accuracy.

References [8] collected about 13,000 comments on various channels that especially uploaded music videos with the YouTube API and only considered English comments. They labeled the comments heuristically by assigning a value of zero to one. To derive accurate result, N-gram analysis was used with the classification algorithms. They used Multinomial Naive Bayes, Random Forest, and Support Vector Machine algorithms. To evaluate the performance of these models, F1 scores were used. Support Vector Machines and Random Forests achieved 0.9774 and 0.9726 accuracy, respectively.

Authors of [9] collected the original comments of the five most viewed YouTube videos through the YouTube API. The comments of each video were classified as spam or ham manually with the collaborative tagging tool (i.e., Labeling). They used the Bag of Words model (BoW) [10] and Term frequency(TF) techniques on these data. Bag of words is a method of representing a document considering only the frequency of words. It is based on TDM which describes the frequency of words in a matrix, ignoring the order. BoW identifies keywords by frequency but cannot figure out the original sentence and its meaning. Nevertheless, [9] used it in data preprocessing. 10 classification methods were applied -CART, k-NN, LR, NB-B, NB-G, NB-M, RF, SVM-L, SVM-P, and SVM-R. 70% of the dataset was used as training data, 30% as test data, and new data was added for testing with algorithms. Ten classifiers showed more than 90% accuracy and less than 5% as blocked ham. As a result, CART, LR, NB-B, RF, SVM-L, and SVM-R showed a 99.9% confidence level.

The purpose of [11] is to compare the results between the classification model used in [9] and an Artificial Neural Network(ANN). They used the same dataset, labeled comments on five popular videos. After data preprocessing, they applied ANN and five measures (i.e., accuracy rate, spam caught rate, blocked ham rate, F1 measure, and MCC) were given. According to the results from these measures, this research using ANN presented higher accuracy for F1 measure and MCC than [7] with a similar blocked ham rate.

References [12] summarizes and compares the classification techniques, datasets, and results used in six papers on YouTube spam comment detection. All researches used at least two techniques, and as a result, combined machine

learning classifiers show good performance. That is the way to enhance the accuracy of classification.

III. SYSTEM MODEL AND PROPOSED METHOD

A. EXPERIMENT METHOD AND ENVIRONMENT

Our proposed method is based on comparative research [9] which is a representative study on YouTube spam comment detection. Our method applied six machine learning techniques (i.e., CART (Decision Tree), LR (Logistic Regression), NB-B (Bernoulli Naïve Bayes), RF (Random Forest), SVM-L (Support vector machine with linear kernel), and SVM-R (Support vector machine with Gaussian kernel)) to improve the performance of the Cascaded Ensemble Machine Learning Model aware YouTube Spam Comments Detection Scheme. These performed well in [9] and were significant with 99.9% confidence. We propose an ensemble model combining them and evaluate the performance.

The experimental environment used version 3.7.1 of Python and version 0.20.1 of the Cicely Library on Jupiter notebooks [13]–[15].

B. EXPERIMENTAL OVERVIEW OF THE PROPOSED TECHNIQUE

As shown in Figure 1, we collect 1,983 comments and distinguish 1,369 (70% of total) into training data and 587 (30%) into test data. To classify the spam data, we remove stop words such as articles (i.e., the, a, an) and pronouns (e.g., I, you, it). Additionally, in [9], only BoW vectorization was performed. In this paper, TF-IDF vectorization preprocessing is used to solve the issue that BoW may not find significant meaning in a sentence because it appears frequently in other sentences. References [12] suggests that there is no single technique that performs well on all datasets. The ensemble model achieved good performance in [9]. We carry on the experiment with multiple techniques to find the best classification algorithm, using six machine learning algorithms (i.e., CART, LR, NB-B, RF, SVM-L, SVM-R). We use two ensemble models, ESM-H (Ensemble with hard voting) and ESM-S (Ensemble with soft voting), to train and test our dataset. They predict and evaluate the class.

C. DATASETS

We use open datasets, which can be downloaded from [16]. They consist of comment data on five popular music videos provided in [9]. They contain YouTube ID, comment author, date, comment content, and labeled class (0: Ham or 1: Spam). We only use comment content and labeled class. Each training and testing of the five data sets as shown in Table.1 can result in overfitting, where the five classifiers perform well only on that data and do not apply well to comment data in other videos. Therefore, in this paper, to generalize the result, we include all five video's datasets. As shown in Fig. 1, we employ 1,983 comments with 1,369 comments (70%) for training and 587 comments (30%) for testing.

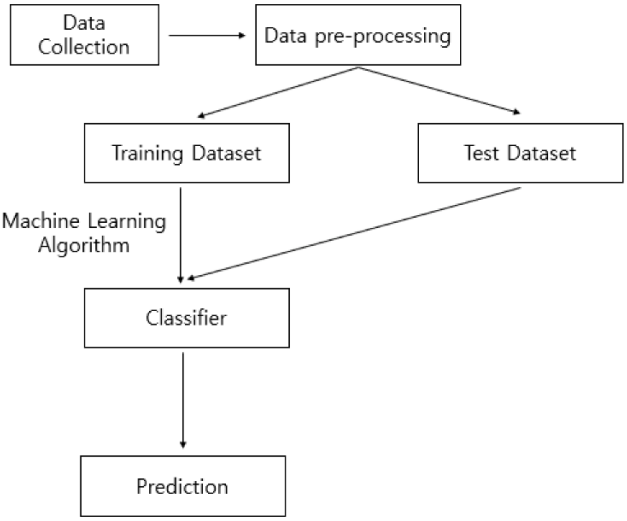

FIGURE 1. Overview of the proposed spam comment detection scheme.

TABLE 1. Datasets collected and used in the experiments.

Datasets	Spam	Ham	Total
Psy	175	175	350
KatyPerry	175	202	350
LMFAO	236	303	438
Eminem	245	203	448
Shakira	174	196	470
Total	1005	978	1983

Comment : "John likes to watch movies. Mary likes movies too."

1. tokenize
["John", "likes", "to", "watch", "movies", "Mary", "likes", "movies", "too"]

2. Bag of words
BoW = {"John":1, "likes":2, "to":1, "watch":1, "movies":2, "Mary":1, "too":1}

FIGURE 2. "Tokenizing and BoW vectorizing processes."

D. DATA PROCESSING

Since the datasets are text data, pre-processing is employed for the machine learning. We eliminate stop words and list the tokens with comments using the CountVectorizer function of the Python Psychic Run library. Then we count how often tokens occur and use BoW (Bag-of-Words) and TF-IDF vectorization. The process is presented in Figure 2.

E. APPLIED MACHINE LEARNING ALGORITHMS

1) DECISION TREE

Decision tree is a method of classification and prediction that uses tree structures to separate entire data into small groups. Separating from a parent node to a child node is called splitting, and the structure of the tree depends on the variables and reference values used in the branch. Variables and reference values with large information gains are selected

by calculating the Gini impurity or Entropy of the parent node and the child nodes. The root node is the topmost node in the tree. It contains all the data to be classified and the group is separated when splitting. Finally, the model is determined by pruning and adjusting the maximum depth of the decision tree. Random forest is a method of making multiple decision trees and outputting the average prediction by changing the training data slightly.

2) LOGISTIC REGRESSION

Logistic regression is a classification algorithm which is mainly used when the categories to be classified are some categories. In particular, logistic regression can be used primarily when the dependent variable is classified as a binary response variable.

For instance, it is used to find the presence or absence of disease according to risk factors such as the incidence of diabetes due to obesity or the response variable. It is also used on the statistical analysis of death or survival with postoperative survival rates that are represented by probability values between 0 and 1.

3) NAÏVE BAYES

Naive Bayes is a supervised learning algorithm that applies Bayes' theory to classification problems. Before the classifier is used, training must be performed with a training vector that calculates the probability that the result will be observed based on the evidence provided by the feature values. It also classifies objects by estimating which class the new data should be included in and predicting them with the highest probability.

The advantages are that it is very efficient in terms of storage space and computation time and handles noise and missing data well. This is because it can include both simple, fast, and evidenced feature vectors. In addition, the training requires relatively few examples because only the number of layers and features indicated while training a case is required, but it works very well for large examples. In particular, since the estimation probability for prediction can be easily obtained, it is well known for its excellent performance in practice. However, all the features of the datasets are equally important and independent, which sometimes makes them unsuitable for practical applications.

4) SUPPORT VECTOR MACHINE

Support Vector Machine is a technique that divides data into groups of similar class values with surface boundaries that create hyperplane boundaries between points that appear in multidimensional space. In other words, the goal of SVM is to create a flat border called a hyperplane that divides space and creates a very homogeneous split on both sides. SVM is a versatile machine learning model because it can be used in various fields such as linear, nonlinear classification, and regression, and is one of the most popular machine learning algorithms.

The principle of dimensional SVM is as follows. The task of the SVM algorithm in two dimensions is to find the maximum margin hyperplane straight line among the multiple dividing straight lines that separate the two classes by the maximum margin. The straightest line that separates with the largest margin will be the most generalized for future data, and the maximum margin increases the likelihood that the point will remain on the right side of the boundary even if there is random noise. In other words, the exact definition of Support Vectors means the points closest to the maximum margin hyperplane MMH in each class. SVM provides a concise way to store classification models, even with a huge number of features and it is easier than using neural networks.

SVMs, on the other hand, must test different combinations of kernel and model parameters to find the best model, which can be slow to train and create complex black box models that are difficult to interpret. Finally, if it is not possible to separate the data linearly, you should use a kernel for nonlinear space to raise the dimension to a separable level. That is, classification between groups that are difficult to classify in two dimensions can be classified in three dimensions.

F. PROPOSED METHODOLOGY

An ensemble model is generating different prediction models using the given data and then combining the results of these prediction models to derive one final prediction result [17].

In this paper, we propose two models using a simple majority vote method. The first one is an ESM-H model that allows more classifiers to adopt the selected class as the final class using a hard voting method. That is, if three classifiers out of five predict class 0 and two predict class 1, an ensemble model would determine the prediction of class 0 with the concept of the hard majority vote. If the number of input classifiers is an even number, it shows the same ratio of the predicted class. To make the number of input classifiers odd, the other five methods, excluding the NB-B which has the lowest performance, are used.

The second model is an ESM-S model, which employs the average of the probabilities of the class predictions from each classifier using a soft voting method. Since each classifier can return the prediction probability of the class, the ESM-S model can finally calculate the average value on top of each classifier with the concept of the soft majority vote.

G. PERFORMANCE EVALUATION METHODS

Various performance evaluation methods have been proposed to understand the effectiveness of the results obtained through the recommendation system. Evaluation methods can be categorized by data type and evaluation purpose. If the data type is continuous, it is evaluated using the accuracy of the recommendation. Categorical data is evaluated with the accuracy of prediction. The purpose of evaluation of the recommendation system can be evaluated by accuracy, unexpectedness, and diversity. The recommendation accuracy is calculated by the difference between the actual preference and the predicted value through an algorithm for predicting the score of the

item, and methods such as Root Mean Squared Error (RMSE) and Mean Average Error (MAE) are used. The classification accuracy is used when evaluating the recommendation performance through the top N items predicted to have high preference, and there are representative methods such as F1 technique, Receiver Operating Characteristic (ROC), precision, and recall [18]–[20]. There are no mathematical formulas to measure diversity and unexpectedness, such as coverage, which is related to how many items are recommended, novelty about the extent of measuring items that are not common to users, and various items. There is a need for studies to mathematically express the concept of diversity related to recommendation.

The purpose of the recommendation system is to suggest recommendation results that are likely to be selected by the user, and to maximize user satisfaction to improve the reliability of the system to ensure continuous use of the system. To evaluate the recommendation, it is necessary to consider accuracy of recommendation performance, psychological factors, and interface elements.

1) EVALUATION METHOD OF THE SCORE PREDICTION ALGORITHM

A score prediction algorithm is generally a method of evaluating the difference between the prediction score and the actual score. The most commonly used valuation scale is MSE, which is a method of obtaining an average value by squaring the difference between each prediction score and the actual score. The equation is as follows, where N is the total number of data, p_{ij} is the predicted score, and r_{ij} is the actual score. MSE is the square of the error and is a method of giving a higher weighting value for a large error.

$$\text{MSE} = \frac{1}{N}\sum \left(p_{ij} - r_{ij}\right)^2$$

RMSE is the evaluation method used in Netflix.

$$RMSE = \sqrt{\frac{1}{N}\sum \left(p_{ij} - r_{ij}\right)^2}$$

The MAE values are as follows, where N is the total number of subjects, p_{ij} is the predicted score, and r_{ij} is the actual score. The absolute value of the difference between the predicted and actual scores is taken and the sum is divided by the total number of subjects. At this time, all scores have the same weight regardless of the magnitude of the error.

$$MAE = \frac{\sum |p_{ij} - r_{ij}|}{\text{N}}$$

Different types of data can have different score scales, and Normalized mean absolute errors (NMAEs) have been developed to normalize them. NMAE divides the difference between the maximum score and the minimum score of the MAE value to produce a normalized result.

$$NMAE = \frac{MAE}{r_{max} - r_{min}}$$

TABLE 2. Confusion matrix.

True/ Prediction	Recommendation	Non-Recommendation
Purchase	a	b
Non-Purchase	c	d

H. EVALUATION METHOD OF THE ITEM RECOMMENDATION ALGORITHM

If the purpose is not to determine the degree of preference for recommendation results but to classify purchases and non-purchases, product viewing and non-viewing, the evaluation is based on the confusion matrix shown in Table 2.

The most common measure is the Mis classification ratio [20], [21]. The probability of correctly classifying the recommendation and the non-recommendation for all items is expressed as follows.

$$\text{Mis Classification Ratio} = \frac{a+b}{a+b+c+d}$$

The recommendation system is a way that a small number of recommended items are selected from many items. It is similar to the concept of 'information retrieval', so the method in the evaluation of information retrieval performance is used. The well-known methods are Precision and Recall.

$$\text{Precision} = \frac{a}{a+c}$$
$$Recall = \frac{a}{a+b}$$

F-measure is a single value that reflects both precision and recall.

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} = \frac{2}{1/Precision + 1/Recall}$$

As the number of recommended products increases, the recall value increases and the precision value decreases. With this tradeoff, F-measure is used as a measure for evaluating how efficient the classification is. As the value approaches 1, both the Recall and Precision values are high.

1) ACCURACY BASED EVALUATION METHOD

Although assessment based on accuracy is used in various studies, the actual practice favors indicators (e.g., lift, hit rate) that measure the benefits of the recommendation system. In fact, predicting an item with a score of one point as four points and predicting a four-point item as a one-point item affects cost differently, but indicators such as MAE do not show this result. Therefore, the evaluation method using the utility function is proposed and used, and the utility is calculated using the utility matrix based on the difference between the actual score(R) and the predicted score ($\hat{R}$). In the following equation, $P\left(\hat{R}_i, R_j\right)$ is the probability that j is predicted by i, and $U\left(\hat{R}_i, R_j\right)$ is the utility of

predicting j by i.

$$EU = \sum_{1 \le i,j \le 10} U\left(\hat{R}_i, R_j\right) P\left(\hat{R}_i, R_j\right)$$

Hit-rate and hit-rank methods are used to measure the performance of the system. They quantify the behavior of the user selecting the recommended results. Hit rate is the ratio of the number of items selected by the user to the number of recommended items. Hit rank is defined as the average of the inverse of the position of the item i ($1 \le i, j \le 10$) by using the weight of the selected item rank.

$$hit\text{-}rate = \frac{Number\ of\ hits}{n}$$

$$hit\text{-}rank = \frac{1}{n} \sum_{i=1}^{h} \frac{1}{p_i}$$

2) DIVERSITY-BASED EVALUATION METHOD

Even in a successful recommendation system, there is a limit to focusing only on the improvement of accuracy. The recommendation system is less common and improves user satisfaction when recommending various and interesting items. However, in the case of accuracy, it is difficult to reflect non-numerical information and new measurement indicators are needed. Psychological or cognitive indicators, such as user satisfaction, are limited in their mathematical expression.

Coverage refers to the ratio of items recommended through the recommendation system to all items, and the more various kinds of recommended items, the higher the coverage. In the content-based approach, if there are problems such as data scarcity in collaborative filtering, the items that are recommended are very limited, resulting in low coverage.

When collaborative filtering uses algorithms based on statistical models, the amount of training data has a big impact on preference prediction. This is defined as the 'learning rate' and classified into total learning rate, learning rate by item, and learning rate by user. This is a method for determining whether the prediction result is reliable in the case of the data scarcity problem.

If a user is recommended L1 at a specific point in time, and then L2 is recommended, an indicator of L2 divided by the number of items not included in L1 by the number of recommendations is proposed.

$$Diversity\,(L_1, L_2, N) = \frac{|L_2 - L_1|}{N}$$

In the case of the diversity indicator, only the difference between the two recommendation lists can be expressed, which is proposed as an extended specificity indicator. It refers to the ratio of the set of recommended items at one time A_t and the items that do not appear. If we define A_t as an existing recommendation history of a user, we can interpret it as an indicator of how many items it is possible to recommend over time.

$$Novelty\,(L_1, N) = \frac{|L_2 - A_t|}{N}$$

3) OTHER EVALUATION METHODS

In the evaluation method of a recommendation system, it is difficult to measure psychological indicators like user satisfaction and system reliability. Such non-numerical information has limitations in verifying through numerical calculation methods, and can be measured by user evaluation, online evaluation, and offline evaluation that evaluate the performance of the recommendation system. Both user evaluations and online evaluations are evaluated by the users.

The difference between them is in what time users evaluate the recommendation system. User evaluation recommends that the user, at a specific point in time, checks the performance of the recommendation system. Online evaluation is a method of evaluating the behavior shown by the user in situations where the recommendation system is utilized in a real environment. The offline evaluation is a data-based evaluation using historical data and user invitation is not needed. In the case of online content recommendations, the user's system evaluation needs to be measured after receiving the recommendation result and using the content.

A user evaluation is made by inviting users and evaluating them through feedback from users who use the recommendation system. The user evaluation can collect desired information such as usage problems or fitness of recommendation to users who have been invited for evaluation. However, the user evaluation recognizes that the invited user evaluates the recommendation system, and there are disadvantages such as expressing one's opinion clearly or distorting the answer.

Online evaluation is a method of grasping the performance of the recommendation system by observing the user's behavior by applying the actual recommendation system without inviting the user separately. It does not recognize that users are evaluating a recommendation system and is very similar to the method used to measure the effectiveness of a new drug. For accurate evaluation, we prefer to test both the use of the recommendation system for the same user and the case not.

Offline evaluation uses historical data to evaluate the performance of the recommendation system. A well-known offline evaluation is the Netflix Prize Contest. It evaluates how accurately participants make a recommendation system and includes historical data. Offline evaluation has the advantage of standardizing evaluation methods and evaluation items, and there are various evaluation items such as accuracy, coverage, confidence, and novelty.

IV. RESULTS

We divide the datasets with 70% for training data and 30% for test data. Then, 10 machine learning techniques are applied, which are presented in Table 3.

Five measures are used for evaluation, Acc (Accuracy rate), SC (Spam caught rate), BH (Blocked ham rate), F1-score, and MCC (Matthews correlation coefficient). Each formula is based on Table 4.

As a result, the ESM-S model showed the best performance in Acc, SC, F1-score, and MCC, and the ESM-S model showed the second-best results with BH after NB-B

TABLE 3. Classification methods used in the experiments.

Classification Method	
CART	Decision Tree
LR	Logistic Regression
NB-B	Bernoulli Naive Bayes
RF	Random Forest
SVM-L	Support vector machine with linear kernel
SVM-R	Support vector machine with Gaussian kernel
ESM-H	Ensemble with hard voting
ESM-S	Ensemble with soft voting

TABLE 4. Confusion matrix. Positive: spam, Negative: ham.

		Predicted	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

TABLE 5. Experiment results.

Methods	Acc(%)	SC(%)	BH(%)	F1-score	MCC	S.D.
CART	93.53	92.18	5.00	0.935	0.871	1.41
LR	92.33	88.27	3.21	0.923	0.851	1.09
NB-B	85.87	73.62	**0.71**	0.858	0.747	1.17
RF	92.84	89.58	3.37	0.928	0.860	1.09
SVM-L	94.21	93.81	5.36	0.942	0.884	1.13
SVM-R	93.19	89.58	2.86	0.932	0.867	1.22
ESM-H	94.38	92.18	3.21	0.944	0.889	0.98
ESM-S	**95.06**	**93.16**	**2.86**	**0.951**	**0.902**	**0.86**

as shown in Table 5. We evaluated the performance of the classifiers through another evaluation method, the Receiver Operating Characteristic (ROC) curve. The ROC curve is a graph created with the x-axis as the FPR (False Positive Rate; the rate of normal comments being incorrectly predicted as a spam) and the y-axis as a Recall (the rate of spam comments correctly predicted as a spam). Figure 3 shows the final ROC curve created by using the FPR of the eight classifiers on the x-axis and Recall on the y-axis. The area under the ROC curve (AUC) seems correct because the area is close to 1, the higher the TP (True Positive; predicting spam as spam) the higher the FN (False Negative; predicting normal comments as normal). Therefore, the ESM-S model shown with a gray line has the largest area of AUC in most datasets.

V. EXPERIMENT WITH DATASETS OF VARIOUS CATEGORIES

We experimented with a new dataset to evaluate the performance of the proposed model in various categories other than music videos. We tested the proposed model in this paper on spam or normal labeled datasets. The dataset consists of 6,431,471 crawled comments of which 481,334 comments

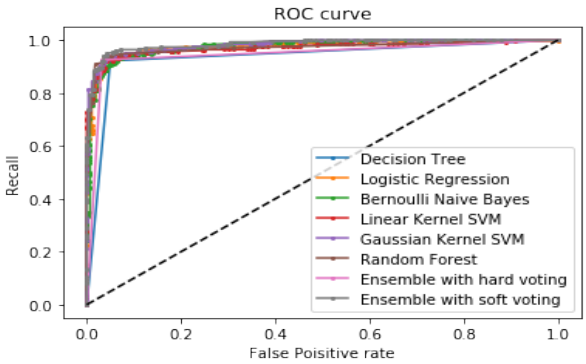

FIGURE 3. ROC curve of the proposed classifiers scheme.

TABLE 6. Experiment results with 1,000 spam and 1,000 ham comments.

Methods	Acc(%)	SC(%)	BH(%)	F1-score	MCC	S.D.
CART	84.67	73.75	4.35	0.845	0.711	1.72
LR	89.00	79.07	1.00	0.889	0.796	1.23
NB-B	87.00	74.42	**0.33**	0.866	0.765	1.34
RF	82.50	84.39	19.40	0.825	0.650	1.31
SVM-L	89.00	82.06	4.01	0.890	0.788	1.32
SVM-R	88.50	78.41	1.34	0.884	0.787	1.24
ANN	89.00	**85.71**	7.69	0.890	0.782	1.41
ESM-H	89.33	80.40	1.67	0.893	0.800	1.01
ESM-S	**90.17**	**83.72**	**3.34**	**0.901**	**0.810**	**0.98**

TABLE 7. Experiment results with 5,000 spam and 5,000 ham comments.

Methods	Acc(%)	SC(%)	BH(%)	F1-score	MCC	S.D.
CART	81.10	67.28	4.73	0.808	0.650	1.92
LR	85.30	77.95	7.16	0.852	0.715	1.53
NB-B	82.13	67.02	**2.36**	0.818	0.677	1.54
RF	84.77	77.88	8.17	0.847	0.703	1.61
SVM-L	85.90	79.99	8.04	0.859	0.724	1.52
SVM-R	86.13	76.70	4.19	0.860	0.737	1.54
ANN	83.50	**83.74**	16.75	0.835	0.670	1.71
ESM-H	86.37	77.68	4.73	0.863	0.738	1.11
ESM-S	**86.43**	78.93	5.87	**0.864**	**0.739**	**1.08**

were spam in the 6,407 videos that were most viewed between October 31, 2011 and January 17, 2012 in the United States. This dataset was mixed with English and non-English comments, so we extracted only English comments for the experiment. In addition, to make it similar to the data size used in the experiment of 3, we extracted 1,000 spam comments and normal comments, and compared them with 5,000 samples. In the experiment, we used an ANN (Artificial Neural Network) technique with the techniques used in 3. Finally, we plotted the Precision, Recall, F1-score, and ROC curves by adding 1,000 data points from 1,000 to 5,000 as shown in Table 6 and Table 7.

Experimental results showed that the ESM-S model performed the best in Acc, F1-score, and MCC in both datasets, and the ANN model in SC, and the NB-B model in BH. In addition, the data set with 1,000 spam comments and 1,000 normal comments performed better than the data set with 5,000 comments.

VI. CONCLUSION

In this paper, we proposed a technique to detect spam comments on YouTube, which have recently seen tremendous growth using a Cascaded Ensemble Machine Learning Model. It examined related studies on YouTube spam comment screening and conducted classification experiments with six different machine learning techniques (Decision tree, Logistic regression, Bernoulli Naïve Bayes, Random Forest, Support vector machine with linear kernel, Support vector machine with Gaussian kernel) and two ensemble models (Ensemble with hard voting, Ensemble with soft voting) combining these techniques in the comment data. The experimental results showed that the ESM-S model proposed in this paper had the best performance in four of five evaluation measures. We proposed a new model, combining various techniques, that improved the performance results unlike previous studies that used one model for detection. We also applied the ensemble model to videos in various categories. It showed that the ESM-S model performed the best in Acc, F1-score, and MCC in both datasets, and the ANN model in SC, and the NB-B model in BH. In addition, the data set with 1,000 spam comments and 1,000 normal comments performed better than the data set with 5,000 comments of the increase in outliers and missing values.

In future research, it is expected that the performance would be better if a TF-IDF or deep learning technique are added.

REFERENCES

[1] S. Aiyar and N. P. Shetty, "N-gram assisted Youtube spam comment detection," *Proc. Comput. Sci.*, vol. 132, pp. 174–182, Jan. 2018, doi: 10.1016/j.procs.2018.05.181.

[2] A. Kantchelian, J. Ma, L. Huang, S. Afroz, A. Joseph, and J. D. Tygar, "Robust detection of comment spam using entropy rate," in *Proc. 5th ACM Workshop Secur. Artif. Intell. (AISec)*, 2012, pp. 59–70, doi: 10.1145/2381896.2381907.

[3] A. Madden, I. Ruthven, and D. Mcmenemy, "A classification scheme for content analyses of Youtube video comments," *J. Documentation*, vol. 69, no. 5, pp. 693–714, Sep. 2013, doi: 10.1108/JD-06-2012-0078.

[4] A. Severyn, A. Moschitti, O. Uryupina, B. Plank, and K. Filippova, "Opinion mining on Youtube," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2014, pp. 1–10, doi: 10.3115/v1/P14-1118.

[5] M. Z. Asghar, S. Ahmad, A. Marwat, and F. M. Kundi, "Sentiment analysis on Youtube: A brief survey," 2015, *arXiv:1511.09142*. [Online]. Available: http://arxiv.org/abs/1511.09142

[6] T. C. Alberto, J. V. Lochter, and T. A. Almeida, "TubeSpam: Comment spam filtering on Youtube," in *Proc. IEEE 14th Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2015, pp. 138–143, doi: 10.1109/ICMLA.2015.37.

[7] A. U. R. Khan, M. Khan, and M. B. Khan, "Naïve multi-label classification of Youtube comments using comparative opinion mining," *Proc. Comput. Sci.*, vol. 82, pp. 57–64, Jan. 2016, doi: 10.1016/j.procs.2016.04.009.

[8] J. Savigny and A. Purwarianti, "Emotion classification on Youtube comments using word embedding," in *Proc. Int. Conf. Adv. Informat., Concepts, Theory, Appl. (ICAICTA)*, Aug. 2017, pp. 1–5, doi: 10.1109/ICAICTA.2017.8090986.

[9] S. Sharmin and Z. Zaman, "Spam detection in social media employing machine learning tool for text mining," in *Proc. 13th Int. Conf. Signal-Image Technol. Internet-Based Syst. (SITIS)*, Dec. 2017, pp. 137–142, doi: 10.1109/SITIS.2017.32.

[10] A. O. Abdullah, M. A. Ali, M. Karabatak, and A. Sengur, "A comparative analysis of common Youtube comment spam filtering techniques," in *Proc. 6th Int. Symp. Digit. Forensic Secur. (ISDFS)*, Mar. 2018, pp. 1–5, doi: 10.1109/ISDFS.2018.8355315.

[11] E. Poche, N. Jha, G. Williams, J. Staten, M. Vesper, and A. Mahmoud, "Analyzing user comments on Youtube coding tutorial videos," in *Proc. IEEE/ACM 25th Int. Conf. Program Comprehension (ICPC)*, May 2017, pp. 196–206, doi: 10.1109/ICPC.2017.26.

[12] A. Aziz, C. F. M. Foozy, P. Shamala, and Z. Suradi, "Youtube spam comment detection using support vector machine and k-nearest neighbor," Tech. Rep., 2018, doi: 10.11591/ijeecs.v12.i2.pp607-611.

[13] R. K. Das, S. S. Dash, K. Das, and M. Panda, "Detection of spam in Youtube comments using different classifiers," in *Advanced Computing and Intelligent Engineering*, 2020, pp. 201–214, doi: 10.1007%2F978-981-15-1081-6_17.

[14] N. M. Samsudin, C. F. B. Mohd Foozy, N. Alias, P. Shamala, N. F. Othman, and W. I. S. Wan Din, "Youtube spam detection framework using Naïve Bayes and logistic regression," *Indonesian J. Electr. Eng. Comput. Sci.*, vol. 14, no. 3, p. 1508, Jun. 2019, doi: 10.11591/ijeecs.v14.i3.pp1508-1517.

[15] G. Kaur, A. Kaushik, and S. Sharma, "Cooking is creating emotion: A study on hinglish sentiments of Youtube cookery channels using semi-supervised approach," *Big Data Cognit. Comput.*, vol. 3, no. 3, p. 37, Jul. 2019, doi: 10.3390/bdcc3030037.

[16] E. Ezpeleta, M. Iturbe, I. Garitano, I. V. de Mendizabal, and U. Zurutuza, "A mood analysis on Youtube comments and a methodfor improved social spam detection," in *Proc. HAIS*, 2018, pp. 514–525, doi: 10.1007/978-3-319-92639-1_43.

[17] N. Hussain, H. Turab Mirza, G. Rasool, I. Hussain, and M. Kaleem, "Predilection decoded: Spam review detection techniques: A systematic literature review," *Appl. Sci.*, vol. 9, no. 5, p. 987, Mar. 2019, doi: 10.3390/app9050987.

[18] L. Song, R. Y. K. Lau, R. C.-W. Kwok, K. Mirkovski, and W. Dou, "Who are the spoilers in social media marketing? Incremental learning of latent semantics for social spam detection," *Electron. Commerce Res.*, vol. 17, no. 1, pp. 51–81, Mar. 2017, doi: 10.1007/s10660-016-9244-5.

[19] S. Jain and D. M. Patel. (2019). *Analyzing User Comments of Learning Videos From Youtube Using Machine Learning*. [Online]. Available: http://www.ijirset.com/upload/2019/august/50_Analyzing_DJ.PDF

[20] P. Bansal. (2019). *Detection of Offensive Youtube Comments, a Performance Comparison of Deep Learning Approaches*. [Online]. Available: https://core.ac.uk/reader/301313034

[21] G. Shi, F. Luo, Y. Tang, and Y. Li, "Dimensionality reduction of hyperspectral image based on local constrained manifold structure collaborative preserving embedding," *Remote Sens.*, vol. 13, no. 7, p. 1363, Apr. 2021, doi: 10.3390/rs13071363.

[22] W. Li and Q. Du, "Laplacian regularized collaborative graph for discriminant analysis of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7066–7076, Dec. 2016, doi: 10.1109/TGRS.2016.2594848.

HAYOUNG OH received the Ph.D. degree in computer science from Seoul University, Seoul, South Korea, in 2013.

From 2001 to 2004, she was a Researcher and a Developer with the Institute of Shinhan Financial Group, Seoul. She was a Visiting Scholar from U.C. Berkeley, in 2010. She was an Assistant Professor of computer science with Soongsil University and Ajou University, from 2014 to 2019. Since 2020, she has been an Associative Professor with the College of Computing and Informatics, Sungkyunkwan University. Her major is artificial intelligence with big data analysis. Her research interests include social network analysis, recommender systems, spam detection, and natural language processing techniques using machine learning and big data analysis.