Feature Selection for Comment Spam Filtering on YouTube

Alper Kursat Uysal [1]
[1]*Department of Computer Engineering, Eskisehir Technical University, Eskisehir, Turkey*

***Abstract*— Spam filtering is one of the most popular domains for text classification. While there exist some many studies on classification of spam e-mails and short text messages, comment spam filtering on YouTube is relatively a new topic as there are limited numbers of annotated datasets. As it is valid for all text classification problems, feature space's high dimensionality is one of the biggest problems for spam filtering due to accuracy considerations. The contribution of this study is the analysis of the performance of five state-of-the-art text feature selection methods for spam filtering on YouTube using two widely-known classifiers namely naïve Bayes (NB) and decision tree (DT). Five datasets including spam comments belonging to different subjects were utilized in the experiments. These datasets are named as Psy, KatyPerry, LMFAO, Eminem, and Shakira. For evaluation, Macro-F1 success measure was used. Also, 3-fold cross-validation is preferred for a fair performance evaluation. Experiments indicated that distinguishing feature selector (DFS) and Gini Index (GI) methods are superior to the other three feature selection methods for spam filtering on YouTube. However, the performance of DT classifier is better than NB classifier in most cases for spam filtering on YouTube.**

***Keywords*—Feature selection, pattern recognition, spam filtering, YouTube.**

I. INTRODUCTION

Text classification, also known as text categorization, has become very popular since the evolution of the Internet. The aim of text classification is to assign electronic documents into pre-defined set of categories. It has various application areas such as sentiment classification [1], medical document classification [2], news classification [3], spam e-mail filtering [4], spam short message filtering [5], and spam comment filtering on social media [6]. Due to the rise in the usage of social medial platforms such as YouTube, Facebook, and Twitter, the number of comment spam increased and comment spam filtering on social media platforms has become popular. Although there exist many studies dealing with spam e-mail and spam short message filtering, comment spam filtering on social media platforms is one of the recent and popular domains in text classification. In the following paragraphs, a literature review is given about comment spam filtering on social media platforms as this study specifically focuses on comment spam filtering on YouTube.

Serbanoiu and Rebedea proposed a ranking mechanism in order to assess the relevance of each comment on YouTube for the individual video [7]. Initially, they collected the first 100 comments for each video by using YouTube Data API. Then, they removed comments written in a language different from English. Besides, they performed topic extraction for individual comments using the Wordnet and Mallet library before classification of comments using neural network. They concluded that their two-step method can be used to construct a good relevance ranking tool for YouTube video comments. Radulescu et al. constructed a system to detect comment spams by applying natural language processing methods and machine learning approaches [8]. They utilized a benchmark dataset and analysed the performances of three classifiers namely decision tree, Support Vector Machines, and naïve Bayes on this dataset. Therefore, the experiments were performed from the data corpus collected from the Daily Telegraph and YouTube. According to experimental results, they stated that best results were obtained by using the decision tree classifier. Alberto et al. evaluated several classifiers for comment spam filtering on YouTube [6]. They stated that the success ratio of some classifiers such as decision trees, naïve Bayes, random forests, logistic regression, Support Vector Machines are nearly equivalent to each other according to statistical analysis. Then, they proposed the tool TubeSpam which is a successful online system to detect comment spams posted on YouTube. For the experiment, they collected data from YouTube and created five datasets by extracting data from YouTube. They concluded that the tool TubeSpam achieved good results with accuracy rates around 95% in the training phase. Alsaleh et al. proposed a comment spam detection system that can be installed as a plugin for web browsers and remove comments including spam content [9]. For the experiments, they manually labelled a new dataset consisting of blog comments that include spam. Four classifiers were utilized in the experiments. However, they applied two attribute evaluators namely CfsSubsetEval and BestFirst search method from WEKA for feature selection. They concluded that the best results were achieved by neural networks, Support Vector Machines, random forest tree, and decision tree classifiers either using all features or subset of the features. Zaman and Sharmin employed several classification algorithms to detect spam comments in YouTube video comments [10]. These classification algorithms are naïve Bayes, k-nearest neighbour, Support Vector Machines, and an ensemble classifier namely bagging. Five datasets [6] collected in a previous study were used for the assessment. They stated

Manuscript received November 29, 2018; accepted December 20, 2018.
*Corresponding author: akuysal@anadolu.edu.tr

that naïve Bayes and bagging classifier give higher accuracy than others in most of the cases. Abdullah et al. compared the performance of nine classification algorithms for YouTube comment spam detection [11]. In the experiments, the used data extracted from YouTube using YouTube Data API. They reported that the best accuracy they obtained is 99.11% and it is obtained with adaptive genetic algorithm. Aiyar and Shetty analysed the performance of n-gram approaches for YouTube spam comment detection [12]. They applied some classification algorithms such as Support Vector Machines, random forest tree, and naïve Bayes on the data collected using Youtube API. They reported that the performances of character-grams are better than word-grams. Besides, random forest tree and Support Vector Machines classifiers are more successful than naïve Bayes classifier.

It should be noted that there exist limited number of studies performed for comment spam detection on YouTube. According to the literature, most of the researchers collected data by themselves and created their own dataset to carry out experiments. Feature selection was not applied in most of the studies. However, feature selection methods which are not specific to text classification were employed in some of the studies. It should be noted that some studies utilize feature selection methods proposed for general pattern recognition problems rather than the ones specific to text classification. This study aims to make an extensive performance analysis on five recently published public datasets namely Psy, KatyPerry, LMFAO, Eminem, and Shakira for detecting YouTube comment spams. Therefore, the performances of two widely-known classifiers namely naïve Bayes and decision tree were assessed using five text feature selection approaches on these datasets.

The flow of the paper is as follows. The feature selection methods utilized in the experiments are explained in Section 2. Classification algorithms applied in the study are describes in Section 3. Results of the experiments are presented in Section 4 and some concluding statements are given in Section 5.

II. FEATURE SELECTION METHODS

Univariate filter-based feature selection methods are widely preferred for text classification as there exist high number of features and these kind of methods not interact with classifiers during feature selection process. Therefore, five well-known univariate filter-based text feature selection approaches are employed in the study. These are information gain [13], Gini index [14], distinguishing feature selector [15], discriminative features selection [16], and relative discriminative criterion [17]. Theoretic backgrounds of these methods are given in the next parts of this section.

A. Information gain (IG)

IG calculates the influence ratio of the absence or presence of a specific term to correct classification decision [18]. Information gain based feature selection can be implemented for text classification as below. If IG score for a term is high, it means that the corresponding term is discriminative. However, IG is a global feature selection method producing a unique score for a term.

$$IG(t) = -\sum_{i=1}^{M} P(C_i)\log P(C_i) + P(t)\sum_{i=1}^{M} P(C_i \mid t)\log P(C_i \mid t) + P(\bar{t})\sum_{i=1}^{M} P(C_i \mid \bar{t})\log P(C_i \mid \bar{t}), \quad (1)$$

In the formula, M represents the class count and $P(C_i)$ represents the probability for class C_i. While $P(\bar{t})$ and $P(t)$ represent the probabilities regarding absence and presence of term t, $P(C_i \mid t)$ and $P(C_i \mid \bar{t})$ represent the probabilities for class C_i when term t is present and absent, respectively.

B. Gini index (GI)

Gini index is a kind of node splitting criteria used in the construction of decision trees [14]. However, GI is an improved version of this criteria and it produces a unique score for each term. It can be formulated as below.

$$GI(t) = \sum_{i=1}^{M} P(t \mid C_i)^2 P(C_i \mid t)^2 \quad (2)$$

While $P(C_i \mid t)$ represent the probability of class C_i inside the documents term t occur, $P(t \mid C_i)$ is the probability of term t inside the documents of class C_i.

C. Distinguishing feature selector (DFS)

DFS is a filter-based feature selection techniques for text classification [15]. It is a global selection method producing a unique score for each term. It can be formulated as below.

$$DFS(t) = \sum_{i=1}^{M} \frac{P(C_i \mid t)}{P(\bar{t} \mid C_i) + P(t \mid \bar{C}_i) + 1} \quad (3)$$

In the formula, $P(C_i \mid t)$ represents the probability of class C_i inside the documents term t occur. M represents class count and $P(\bar{t} \mid C_i)$ is the conditional probability of lack of term t inside the documents labelled as class C_i. However, $P(t \mid \bar{C}_i)$ is the probability of term t when all other classes except C_i present.

D. Discriminative features selection (DFSS)

DFSS [16] is one of the recent univariate filter-based text feature selection methods aiming to select features with a high document frequency and high average term frequency inside documents of a specific class. It can be formulated as below.

$$DFSS(t,C) = \frac{\mathrm{tf}(t,C)/\mathrm{df}(t,C)}{\mathrm{tf}(t,\bar{C})/\mathrm{df}(t,\bar{C})} \times \frac{a}{(a+b)} \times \frac{a_i}{(a+c)} \times \left| \frac{a}{(a+b)} - \frac{c}{(c+d)} \right| \quad (4)$$

In the formula, $\mathrm{tf}(t,\bar{C})$ and $\mathrm{tf}(t,C)$ are the frequency of feature t in other categories and category C, respectively. $\mathrm{df}(t,C)$ is the number of text documents belonging to category C including feature t. $\mathrm{df}(t,\bar{C})$ is the number of text documents for other categories including feature t.

While a is the number of text documents in category C including feature t, b is the number of text documents in category C not including feature t. While c is the number of text documents in categories except C including feature t, d is the number of text documents in the categories except category C not including feature t. Class-based feature scores are globalized using maximum globalization function.

E. Relative discriminative criterion (RDC)

RDC is a new method considering document frequencies for each term count of a term [17]. It is not a feature selection method relying on probability like most of the other filter-based approaches. The flow of RDC algorithm can be shown as below. Class-based scores for features are obtained with this algorithm and they are globalized using weighted average globalization function.

Algorithm 1. The flow of RDC algorithm

POS is the amount of documents inside positive class
NEG is the amount of documents inside negative class
TCMAX is the maximum term count for term t
tp_{tc} is the amount of positive documents including term t with term count tc
fp_{tc} is the amount of negative documents including term t with term count tc

for tc = 1 to TCMAX do
tpr_{tc} = tp_{tc} / POS
fpr_{tc} = fp_{tc} / NEG
D_{tc} =| tpr_{tc} - fpr_{tc} |

$$RDC_{tc} = \frac{D_{tc}}{\min(\mathrm{tpr}_{\mathrm{tc}}, \mathrm{fpr}_{\mathrm{tc}}) * \mathrm{tc}}$$

end
AUC_{tc} = 0
for tc = 1 to TCMAX do

$$AUC_{tc} = AUC_{tc} + \frac{RDC_{tc} + RDC_{tc+1}}{2}$$

end

III. CLASSIFICATION ALGORITHMS

Two widely-known classifiers are utilized to examine the effectiveness of the selected features. These classifiers are naïve Bayes (NB) and decision tree (DT). The statements in the next subsections explain these two classifiers.

A. Naïve Bayes (NB)

NB is a well-known classifier relying based on Bayes theorem. NB classifier assumes that the features do not correlate with each other. Therefore, a probability score is calculated with multiplication of some conditional probabilities. Multi-variate Bernoulli and multinomial event models are known as successful event models for NB classifier and they are specific to text classification [19]. Multi-variate Bernoulli event model is utilized in this study while implementing naïve Bayes classifier.

B. Decision tree (DT)

DT classifier aims to reach a classification decision with the help of a decision tree structure it constructed. Nodes in the decision tree structure generally represent feature values and leaves in the decision tree structure represent specific class labels. When a new test sample is given to DT classifier as input, decisions are made depending on the feature values of the new test sample. The final leaf node that can be reached on the decision tree structure will be the target class label for the new sample.

IV. EXPERIMENTAL WORK

In this section, a comprehensive analysis was attained to compare the performances of five filter-based feature selection methods using two different classifiers. Term weighting is performed using TF-IDF method. In the rest of the section, some details about employed datasets and used success measure are given besides presenting experimental results.

A. Datasets

Five recently published public datasets [6] were utilized in this study. More information about these datasets are presented in Table I.

TABLE I. DATASETS UTILIZED IN THE STUDY

Dataset Name	Spam	Ham	Total
Psy	175	175	350
KatyPerry	175	175	350
LMFAO	236	202	438
Eminem	245	203	448
Shakira	174	196	370

B. Success measure

In the study, Macro-F1 measure [20] is used for evaluation. Macro-F1 score considers induvial classification performances of classes. Macro-F1 measure can be formulated as below.

$$Macro - F1 = \frac{\sum_{k=1}^{C} F_k}{C}, \qquad F_k = \frac{2 \times p_k \times r_k}{p_k + r_k}, \tag{5}$$

In the formula, p_k and r_k are precision and recall scores for class *k*, respectively.

C. Accuracy analysis

In this section, the performances of five text feature selection methods were assessed using Macro-F1 score. The selected features were fed into NB and DT classifiers as input. In the experiments, different feature sizes were used. Lowercase conversion and Porter stemming [21] were used in addition to weighting terms with TF-IDF. For a fair performance evaluation, 3-fold cross-validation is used in the experiments. Macro-F1 scores achieved on these five datasets are listed in Tables II-VI. In the tables, the highest Macro-F1 scores for each dataset and classifier are indicated in bold.

TABLE II. MACRO-F1 SCORES (%) FOR PSY DATASET USING NB AND DT CLASSIFIERS

	NB					DT				
Feature Size	10	50	100	200	300	10	50	100	200	300
IG	92.46	93.32	**94.48**	93.59	92.41	91.29	93.92	93.92	93.92	93.92
GI	92.15	93.32	**94.48**	93.59	92.43	91.87	93.92	93.92	93.92	93.92
DFS	91.55	92.73	93.62	93.30	92.70	90.67	93.92	93.92	93.92	93.92
DFSS	83.16	83.58	84.22	86.83	86.59	83.22	80.24	80.02	84.11	85.04
RDC	80.44	83.58	84.52	86.53	86.89	80.16	80.55	80.70	84.11	85.68

TABLE III. MACRO-F1 SCORES (%) FOR KATYPERRY DATASET USING NB AND DT CLASSIFIERS

	NB					DT				
Feature Size	10	50	100	200	300	10	50	100	200	300
IG	91.86	92.44	90.08	88.26	87.93	91.30	92.75	92.45	92.45	92.45
GI	88.00	92.44	89.78	88.28	87.95	87.75	92.75	**93.32**	92.45	92.45
DFS	91.85	92.73	90.68	88.87	87.64	90.68	93.04	92.16	92.45	92.45
DFSS	89.19	88.39	88.63	88.34	89.19	88.67	89.56	89.56	89.56	91.00
RDC	88.95	87.22	88.05	89.81	89.19	90.40	89.84	90.13	92.15	91.57

TABLE IV. MACRO-F1 SCORES (%) FOR LMFAO DATASET USING NB AND DT CLASSIFIERS

	NB					DT				
Feature Size	10	50	100	200	300	10	50	100	200	300
IG	91.00	92.84	93.29	93.74	93.29	93.08	96.06	96.06	96.06	95.60
GI	91.22	93.07	93.77	93.74	93.29	93.07	96.06	96.06	96.06	95.60
DFS	91.46	92.38	93.29	93.06	93.76	93.31	**96.29**	96.06	96.06	95.60
DFSS	78.81	81.46	81.02	86.12	86.83	76.69	78.18	77.72	87.76	87.76
RDC	77.10	80.56	84.74	86.12	86.59	75.41	77.10	84.52	87.76	87.76

TABLE V. MACRO-F1 SCORES (%) FOR EMINEM DATASET USING NB AND DT CLASSIFIERS

	NB					DT				
Feature Size	10	50	100	200	300	10	50	100	200	300
IG	94.13	93.00	90.75	87.60	89.85	93.00	**96.61**	96.61	95.92	95.70
GI	93.45	92.33	90.30	88.05	88.72	93.46	**96.61**	96.61	95.92	95.70
DFS	94.35	90.08	90.52	87.37	90.08	93.68	**96.61**	95.92	95.92	95.70
DFSS	67.82	75.17	74.27	86.71	86.92	65.87	70.67	71.56	85.34	85.34
RDC	67.25	74.73	74.73	86.48	86.70	66.87	71.30	71.04	85.12	85.34

TABLE VI. MACRO-F1 SCORES (%) FOR SHAKIRA DATASET USING NB AND DT CLASSIFIERS

	NB					DT				
Feature Size	10	50	100	200	300	10	50	100	200	300
IG	88.82	83.76	79.45	73.79	70.67	89.37	91.11	90.03	89.76	89.76
GI	89.15	82.58	79.45	73.79	71.03	88.32	**91.67**	90.03	89.76	89.76
DFS	89.45	82.88	79.75	74.47	71.03	88.56	90.57	89.47	89.20	89.76
DFSS	66.50	67.42	65.42	79.07	75.16	64.93	72.66	71.42	82.15	81.98
RDC	63.92	67.29	65.07	79.07	74.46	61.97	71.66	71.95	82.15	82.29

The best Macro-F1 value for NB classifier were achieved using IG and GI feature selection methods with 100 features on Psy dataset according to Table 2. However, the best Macro-F1 value for DT classifier was achieved using IG, GI, and DFS feature selection methods with 50 features. The performance of NB classifier is better than DT according to the highest Macro-F1 scores. The performances of DFSS and RDC are worse than the other feature selection methods in general for Psy dataset.

The best Macro-F1 value for NB classifier was achieved using DFS feature selection methods with 50 features on KatyPerry dataset according to Table 3. However, the best

Macro-F1 value for DT classifier was achieved using GI feature selection method with 100 features. The performance of DT classifier is better than NB according to the highest Macro-F1 values.

The best Macro-F1 score for NB classifier were achieved using GI feature selection method with 100 features on LMFAO dataset according to Table 4. However, the best Macro-F1 value for DT classifier was achieved using DFS feature selection method with 50 features. The performance of DT classifier is better than NB according to the highest Macro-F1 values. The performances of DFSS and RDC are worse than the other feature selection methods in general for LMFAO dataset.

The best Macro-F1 value for NB classifier was achieved using DFS feature selection method with 10 features on Eminem dataset according to Table 5. However, the best Macro-F1 value for DT classifier was achieved using IG, GI, and DFS feature selection method with 50 features. The performance of DT classifier is better than NB according to the highest Macro-F1 values. The performances of DFSS and RDC are worse than the other feature selection methods in general for Eminem dataset.

The best Macro-F1 value for NB classifier was achieved using DFS feature selection method with 10 features on Shakira dataset according to Table 6. However, the best Macro-F1 value for DT classifier was achieved using GI feature selection method with 50 features. The performance of DT classifier is better than NB according to the highest Macro-F1 values. The performances of DFSS and RDC are worse than the other feature selection methods in general for Shakira dataset.

When overall highest Macro-F1 values obtained on five datasets are considered, DFS and GI methods are superior to the other three feature selection methods for spam filtering on YouTube. However, DT classifier is more successful than NB classifier in most of the cases. Most of the highest accuracies were obtained with 50 or 100 features. The performances of DFSS and RDC are worse than the other feature selection methods for some of the datasets.

V. CONCLUSIONS

In this study, the performance of five successful text feature selection methods in the literature were examined for spam comment filtering on YouTube. NB and DT classifiers were utilized to test the efficacy of these approaches. Five recently published datasets namely Psy, KatyPerry, LMFAO, Eminem, and Shakira were utilized in the experiments. Macro-F1 success measure was used in this study. Experiments indicated that most of the highest classification performances were attained with DFS and GI feature selection methods. However, DFSS and RDC seem less effective in comparison to the others for some of the cases. As a future work, the performances of these successful text feature selection algorithms can be analyzed with various different classifiers.

REFERENCES

[1] A. K. Uysal and Y. L. Murphey, "Sentiment classification: Feature selection based approaches versus deep learning," in *17th IEEE International Conference on Computer and Information Technology (CIT)*, 2017, pp. 23-30.

[2] B. Parlak and A. K. Uysal, "The impact of feature selection on medical document classification," in *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)*, 2016, pp. 1-5.

[3] B. K. Akkuş and R. Cakici, "Categorization of Turkish news documents with morphological analysis," in *51st Annual Meeting of the Association for Computational Linguistics Proceedings of the Student Research Workshop*, 2013, pp. 1-8.

[4] P. P. K. Chan, C. Yang, D. S. Yeung, and W. W. Y. Ng, "Spam filtering for short messages in adversarial environment," *Neurocomputing*, vol. 155, pp. 167-176, 2015.

[5] A. K. Uysal, S. Gunal, S. Ergin, and E. S. Gunal, "The impact of feature extraction and selection on SMS spam filtering," *Elektronika ir Elektrotechnika (Electronics and Electrical Engineering)*, 2013.

[6] T. C. Alberto, J. V. Lochter, and T. A. Almeida, "Tubespam: Comment spam filtering on YouTube," in *Machine Learning and Applications (ICMLA), 2015 IEEE 14th International Conference on*, 2015, pp. 138-143: IEEE.

[7] A. Serbanoiu and T. Rebedea, "Relevance-based ranking of video comments on YouTube," in *Control Systems and Computer Science (CSCS), 2013 19th International Conference on*, 2013, pp. 225-231: IEEE.

[8] C. Rădulescu, M. Dinsoreanu, and R. Potolea, "Identification of spam comments using natural language processing techniques," in *Intelligent Computer Communication and Processing (ICCP), 2014 IEEE International Conference on*, 2014, pp. 29-35: IEEE.

[9] M. Alsaleh, A. Alarifi, F. Al-Quayed, and A. Al-Salman, "Combating comment spam with machine learning approaches," in *Machine Learning and Applications (ICMLA), 2015 IEEE 14th International Conference on*, 2015, pp. 295-300: IEEE.

[10] S. Sharmin and Z. Zaman, "Spam detection in social media employing machine learning tool for text mining," in *Signal-Image Technology & Internet-Based Systems (SITIS), 2017 13th International Conference on*, 2017, pp. 137-142: IEEE.

[11] A. O. Abdullah, M. A. Ali, M. Karabatak, and A. Sengur, "A comparative analysis of common YouTube comment spam filtering techniques," in *Digital Forensic and Security (ISDFS), 2018 6th International Symposium on*, 2018, pp. 1-5: IEEE.

[12] S. Aiyar and N. P. Shetty, "N-Gram assisted Youtube spam comment detection," *Procedia Computer Science*, vol. 132, pp. 174-182, 2018.

[13] A. K. Uysal, "An improved global feature selection scheme for text classification," *Expert Systems with Applications*, vol. 43, pp. 82-92, 2016.

[14] W. Shang, H. Huang, H. Zhu, Y. Lin, Y. Qu, and Z. Wang, "A novel feature selection algorithm for text categorization," *Expert Systems with Applications*, vol. 33, no. 1, pp. 1-5, 2007.

[15] A. K. Uysal and S. Gunal, "A novel probabilistic feature selection method for text classification," *Knowledge-Based Systems*, vol. 36, pp. 226-235, 2012.

[16] W. Zong, F. Wu, L.-K. Chu, and D. Sculli, "A discriminative and semantic feature selection method for text categorization," *International Journal of Production Economics*, vol. 165, pp. 215-222, 2015.

[17] A. Rehman, K. Javed, H. A. Babri, and M. Saeed, "Relative discrimination criterion – A novel feature ranking method for text data," *Expert Systems with Applications*, vol. 42, no. 7, pp. 3670-3681, 2015.

[18] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *Journal of Machine Learning Research*, vol. 3, pp. 1289-1305, 2003.

[19] L. Jiang, Z. Cai, H. Zhang, and D. Wang, "Naive Bayes text classifiers: A locally weighted learning approach," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 25, no. 2, pp. 273-286, 2013/06/01 2013.

[20] A. K. Uysal, "On two-stage feature selection methods for text classification," *IEEE Access*, vol. 6, pp. 43233-43251, 2018.

[21] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130-137, 1980.