Buzzer Detection on Indonesian Twitter using SVM and Account Property Feature Extension

Yuliant Sibaroni[1], Sri Suryani Prasetiyowati[2]
[1,2]School of Computing, Telkom University
[1]yuliant@telkomuniversity.ac.id, [2]srisuryani@telkomuniversity.ac.id

Abstract

The rapid use of Twitter social media in recent times has an impact on the faster dissemination of disinformation which is very dangerous to followers. Detection of disinformation is very important to do and can be done manually by conducting in-depth information analysis. But given the huge amount of information, this approach is less effective. Another, more effective approach is to use a machine learning-based approach. Several studies on hoax information detection based on machine learning have been carried out where some studies analyze the content of a tweet and some others analyze hashtags which are the context of a tweet. The feature usually used to analyze hashtag sentiment data is the property feature of the creator's account. The creator accounts of disinformation are called buzzer accounts. This research proposes account property feature expansion of buzzer accounts combined with the SVM classifier which in several previous similar studies has a very good performance to detect the buzzer hashtag. The experimental results show that expanding the proposed feature can increase SVM's performance in detecting hashtag buzzers by more than 24% compared to using the baseline feature, and the average F1 score obtained from the combination of methods is 84%.

Keywords: Twitter, account, hoax, buzzer, SVM

1. Introduction

The internet is currently growing rapidly in Indonesia where data shown by [1] shows that in January 2021 around 170 million people in Indonesia have accessed the internet and social media applications. Social media applications are one of the most effective media in disseminating information widely. In 2020, the 5 most used social media in Indonesia are Youtube, Whatsapp, Facebook, Instagram, and Twitter [2].

At the beginning of the presence of modern online social media such as Facebook (2004), Twitter (2006), Instagram (2010), and other social media, issues related to hoax information in social media have not become an important issue. But at this time, when the internet is growing rapidly and most people already have personal social media, the issue of handling hoax information on social media is very important.

Studies on fake news and hoax information on various social media platforms have become a research topic that has attracted many researchers to date, as was conducted by [3] on Facebook, [4] on Instagram, and also by [5] on Twitter. Studies using social media data are mostly done on Twitter social media, this is due to

the ease of crawling research data on Twitter in general, it is easier for the public to do than crawling other social media data. Another reason is that the level of hoax information on Twitter is higher than on other social media. Research conducted by [6] shows that from 2015-2018 the trend of fake news sites on Twitter showed an increasing trend, while Facebook showed a declining trend even though it had an upward trend at the beginning of that period. The emergence of fake news and hoax information is generally caused by the presence of a buzzer and also fake accounts or robots that are intended to produce invalid information.

At first, the term buzzer came from the marketing field, namely buzz marketing or a technique of marketing goods or services to generate business by moving information by word of mouth [7]. In online social media, a buzzer was originally defined as a social media account whose job is to disseminate, campaign, and broadcast a message or content to amplify a message so that the content becomes public opinion [8]. But in recent times, the term buzzer has shifted to a social media account that has a large number of followers and participates in political campaigns by spreading various hoax news and hate speech to the opposing party

[4],[9],[10]. In a study conducted by Bradshaw [11] buzzer in Indonesia itself has been categorized as a low cyber troop army. This means that buzzers in Indonesia often carry out disinformation and spread the disinformation widely through certain hashtags created by them. The effect of the buzzer in Indonesia has reached such a bad stage that it can divide society [9].

Research on the detection of bot accounts or robots on Twitter was conducted by [12]. This research proposes a bot account classification system based on entropy components, spam detection components, account properties components, and classification based on decision-makers. The features used are extracted from unknown users. The latest development, account detection on Twitter is not only limited to bot account detection but develops towards detecting buzzer accounts on Twitter [13][14]. Regarding buzzer detection in elections, there are several similar studies on Twitter social media conducted by [5], [10], [13]–[15] where the problem domain is buzzer detection in elections. Some important findings obtained from some of these researches include bot accounts having a lower ratio of the number of followers or followings compared to real accounts, posting times made by bot accounts are more regular, and bot accounts, in general, have a younger account age than the original account. In detecting buzzer accounts, the original user metadata and activity features are generally used, both on Instagram [4] and Twitter [5], [16]. But considering the phenomenon of the emergence of buzzer accounts not only in these periods, but research on the detection of the buzzer account in different periods is also interesting to do. Another interesting thing to be developed is in features used so far, namely original user metadata and activity features.

In this paper, buzzer hashtag classification is carried out based on the property of the buzzer account. Data containing hashtag buzzer is retrieved not only in the election period. The classifier method used is the Support Vector Machine (SVM). The use of SVM is based on the fact that this classifier has a fairly good performance in text classification based on several previous studies [17]–[19]. SVM model has several kernel models and parameter variations such as C and epsilon where a lot of experiments are needed to obtain optimal results. In this research, optimization (tuning) of SVM kernel parameters is carried out. Tuning techniques are also widely used in several Data Mining classifiers from various researches, such as [20]–[23]. The SVM model tuning technique in this research is implemented using several Python language functions in the Scikit-Learn library, namely SVC, NuSVC, and LinearSVC (LSVC).

The contribution of this research is the property feature expansion model of the buzzer account on Twitter based on descriptive statistical measures, Another contribution is knowing the effectiveness of SVM in the process of detecting buzzer accounts on Twitter social media. The process of detecting the buzzer account is conducted by classifying the hashtags created by the buzzer account as natural hashtags (non-buzzer) or buzzer hashtags.

2. Research Methods

There are several stages carried out in this research to achieve the research objectives including data crawling, data labeling, account property feature extraction, account property feature expansion, data normalization, SVM Classification and Evaluation.

2.1 Data Crawling

In this stage, the process of collecting research data from Twitter social media data is carried out. In Twitter, the data crawling process is carried out by downloading account data and their properties as well as the contents of tweets from the Twitter server using the Twitter Application Programming Interface (API).

Data crawling was carried out during April 2021 - September 2021. In this research, 202 thousand tweets were successfully crawled from 202 hashtags. Each hashtag contains 1000 tweets which are hashtags that are trending topics at that time. Some examples of hashtags crawled in this research include #Adzanbukamainan, #Hajat4nAsetku, #KRINanggala402, #ReshufflePresident, #17an, and others.

In this crawling process, information related to a Twitter account is obtained such as user id, the time when the account was created, Tweet content, and others. There are 17 types of information that have been extracted from the crawling process. All this information can be seen in Table 1.

Table 1. Data Crawling Process Results

No	Information	Explanation
1	Created	Tweet posting time
2	Id_Tweet	Id of tweet
3	Text	Content of tweet
4	User_Id	The id of user who posted the tweet
5	Username	Username of user
6	User_Screen_Name	Account name of the user
7	Status_Count	the number of posts
8	Verified	Verified account or not
9	Follower	number of followers of an account
10	Following	number of following an account
11	Reply_Tweet_Id	id_tweet of replied tweet
12	Reply_User_Name	username of replied tweet
13	Reply_User_Id	user_id of replied tweet
14	Entities	additional things to tweets such as hashtags, urls, and mentions
15	Url	the link in the tweet
16	User_Created_At	user account creation time
17	Sum_Mention	Number of accounts mentioned in one tweet

2.2 Data Labelling

This process is carried out to determine the label of a hashtag on Twitter as a buzzer hashtag or a non-buzzer hashtag (natural). Labeling was carried out by 3 respondents where each respondent was asked to determine the label of each hashtag. Before labeling, respondents gain knowledge about the definition of buzzer from several references that will be used as knowledge for respondents to determine the hashtag label. Some property information from the account that created the hashtag was also given to the respondents. An example of a hashtag buzzer labeling form can be seen in Figure 1.

Figure 1. Example of Hashtag Labelling Form

The result of hashtag labeling is that 63% of hashtags are labeled the same by the three respondents, while the rest are labeled by a majority vote. The labeling results are balanced as shown in Figure 2.
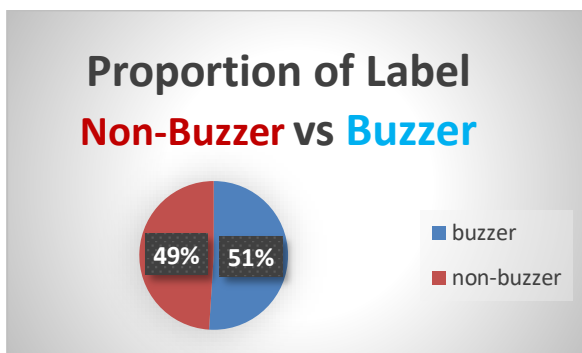

Figure 2. The proportion of Hashtag Label

Some examples of hashtags categorized as buzzer hashtags and natural hashtags can be seen in Table 2.

Table 2. Example Non-Buzzer Hashtag and Buzzer Hashtag

No	Non-Buzzer Hashtag	No	Hashtag Buzzer
1	#Greyapfinaltokyo2020	1	#Australiaberbohong
2	#Happysunshinedae	2	#Bantuanpaketdata
3	#Haribatiknasional	3	#Batalkankartuvaksin
4	#Hospitalplaylist2Ep12	4	#Batalkanpemindahanibukota
5	#Hut76Tni	5	#Bebaskanyahyawaloni

2.3. *Account Property Feature Extraction*

This process is carried out to generate features based on information obtained from the data crawling process. Not all of the information obtained in Table 2 can be used as a feature because in general, the generated features must-have characteristics that can direct whether a hashtag is a buzzer or not. For example, information such as id_tweet, user_id, username are some examples of information that cannot be used as attributes because the information is unique so it cannot lead to a particular hashtag pattern.

Tabel 3. Features of Account Property

No	Feature	Explanation
1	Unique_User_Pertweet	Number of different users in 1000 tweets
2	Url_Pertweet	Number of urls in 1000 tweets
3	Reply_Pertweet	Number of replies in 1000 tweets
4	Reply_User_Pertweet	The number of users who replied in 1000 tweets
5	Rt_Pertweet	Number of ReTweets (RT) in 1000 tweets
6	Unique_Rt_Pertweet	Unique RT count in 1000 tweets
7	Verified_User_Pertweet	Number of verified accounts in 1000 tweets
8	Duration	The difference between the maximum time and the minimum time on the "created" attribute
9	Mean_Mention	Mean of Sum_mention attribute
10	Mean_Number_In_Uname	Mean of number in username attribute
11	Reputation	the ratio between followers and following+followers
12	Ratio	the ratio between followers and following
13	Age_Account	Difference between "created" and "User Created_at" attribute (month)
14	Follower	number of followers of an account
15	Following	number of following of an account
16	Status_count	The number of posts an account
17	Follower_Rate	Follower divided by age_account
18	Following_Rate	Following divided by age_account
19	Status_Rate	Status_count divided by age_account

The results of the account property feature extraction can be seen in Table 3. Some simple features extracted can be obtained directly with simple computations from the information obtained from Table 1, meanwhile other features must be generated based on hashtag data containing 1000 tweets that have been collected for each account.

This account property feature extraction process is carried out according to research from [12] where the account property features, namely feature number 1 to

7 in Table 3, are obtained from the results of data extraction of 1000 tweets in each hashtag. This account property feature is also known as a community feature or attribute, which is an attribute that describes the position of an account in its community [16].

2.4 Twitter Account Property Feature Expansion

This is a process to expand the basic account property features to a wider extent using descriptive statistical measures including mean, quartiles (1,2,3), and range. This feature expansion process was carried out on 9 previously obtained features, namely feature number 11 (reputation) to feature number 19 (Status_rate). The results of feature expansion in these 5 statistical measures can be seen in Table 4.

Table 4. Expansion Result Features

| ID | Old Feature | New Expansion Features | | | | |
		Mean	Q1	Q2	Q3	Range
F1	Reputation	Mean_F1	QI_F1	Q2_F1	Q3_F1	Rg_F1
F2	Rasio	Mean_F2	QI_F2	Q2_F2	Q3_F2	Rg_F2
F3	Age	Mean_F3	QI_F3	Q2_F3	Q3_F3	Rg_F3
F4	Follower	Mean_F4	QI_F4	Q2_F4	Q3_F4	Rg_F4
F5	Following	Mean_F5	QI_F5	Q2_F5	Q3_F5	Rg_F5
F6	Status_count	Mean_F6	QI_F6	Q2_F6	Q3_F6	Rg_F6
F7	Follower_Rate	Mean_F7	QI_F7	Q2_F7	Q3_F7	Rg_F7
F8	Following_Rate	Mean_F8	QI_F8	Q2_F8	Q3_F8	Rg_F8
F9	Status_Rate	Mean_F9	QI_F9	Q2_F9	Q3_F9	Rg_F9

The descriptive statistical formula used in the feature expansion process in this study can be seen in equations 1 and 2:

$$Mean\ X = \frac{\sum X_i}{n} \tag{1}$$

$$Range\ X = Q3 - Q1 \tag{2}$$

Q1 is the first quartile, which is a value that has the property that the percentage of sorted data below this value is 25%. Q2: The second quartile represents the middle value of the sorted data, while Q3 is the third quartile which states that the percentage of sorted data below this value is 75%.

2.5 Data Normalization

The normalization process is carried out so that the range of each feature becomes uniform, namely [0,1]. The normalization formula for a feature X used in this research can be seen in equation 3.

$$Normalization(X) = \frac{X - Min}{Max - Min} \tag{3}$$

Where Min is the minimum value and Max is the Maximum value of the feature X.

2.5 SVM Classification

The classifier model used in this study is Support Vector Machine (SVM) model where this model is one of the machine learning algorithms that can separate two different classes, which can group positive and negative classes using a hyperplane as a separating method. The optimal hyperplane is obtained by maximizing the distance between classes or margins. In the case of binary linear classification, the SVM hyperplane equation can be written in the form of equation 4 [24], [25]:

$$f(x) = w^T x + b \tag{4}$$

x is the data attribute vector, w is the coefficient of x and b is the parameter to be searched in order to obtain the optimal model. Class-y classification based on x_i is defined as equation 5:.

$$y = \begin{cases} 1 & f(x_i) \geq 0 \\ -1 & f(x_i) < 0 \end{cases} \tag{5}$$

The explanation of equation (5) is that if the calculation result of $f(x_i)$ for an attribute vector x_i produces a value greater than or equal to 0, then the data x_i will be classified in class "1", and if the calculation of $f(x_i)$ is less than 0 then xi will be classified in class "-1" [26].

The problem in determining the hyperplane is the determination of the optimal hyperplane. This problem is related to the determination of parameter C which represents the cost of the penalty. The value of C must be determined precisely and cannot be too large or small. This Hyperplane optimization equation can be seen in equation (6)

$$min\ \frac{1}{2}\|\omega\|^2 + C \sum_{i=1}^{n} \varsigma_i \tag{6}$$

that fulfills

$$y_i(w^T x_i + b) \geq 1 - \varsigma_i \quad \text{untuk}\ \varsigma_i \geq 0$$

Where, ς_i is slack variable and represents the misclassified sample of the appropriate margin hyperplane.

SVM uses Kernel functions to transform the original data to a higher dimensional vector space. There are some kernel functions in SVM but there are 4 kernel functions that are generally used the most, namely Linear, Polynomial, Radial Basis Function (RBF), and Sigmoid [25], [27]. Kernel formulas and related parameters for the four kernel models can be seen in Table 5.

Table 5. Frequently used SVM Kernel functions and their parameters [28]

No	Fungsi Kernel	Parameter
1	Linear $\langle x, x' \rangle$	C
2	Gaussian Radial Basis Function (RBF) $Exp(-\gamma\|x - x'\|^2)$	C, γ
3	Sigmoid $tanh(\gamma(x, x') + r)$	C, γ, r
4	Polinomial $(\gamma(x, x') + r)^d$	C, γ, r, d

The implementation of SVM in this research uses the python programming language and the scikit-learn machine learning library. The SVM model is implemented in Python using several functions namely SVC, NuSVC, and LinearSVC. All these functions are executed using parameters by default.

SVC, NuSVC and LSVC are capable of performing binary classification as is the case in this research and also support multi-class classification. SVC stands for C-Support Vector Classification which is implemented based on libSVM and has C parameter with a default value = 1. NuSVC stands for Nu-Support Vector Classification. NuSVC. NuSVC is similar to SVC but has different parameters with SVC namely nu with a default value = 0.5. LSVC stands for Linear-Support Vector Classification. LinearSVC is a faster implementation of SVC based on liblinear and only uses linear kernel.

2.6 Performance Evaluation

The performance of each SVM function is calculated using the F1-score. The F1-score was chosen considering that this measure is often used in research on text mining and is quite robust and related to imbalanced data. The F1-score formula is calculated based on the recall and precision values where the precision, recall, and F1 formulas can be seen in equations (7,8,9), namely:

$$Recall = \frac{TP}{FN + TP} \qquad (7)$$

$$Precision = \frac{TP}{FP + TP} \qquad (8)$$

$$F1 = 2 \ x \ \frac{Precision \ x \ Recall}{Precision + \ Recall} \qquad (9)$$

Where information about TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative) can be seen in the confusion matrix for binary classification case in Figure 2.

Figure 2. Confusion Matrix

Positive in this case states that the data has a Buzzer label, while Negative states that the data is labeled as non-buzzer. Based on the confusion matrix, it can be explained that TP is a condition when the predicted result is a buzzer and the actual label is also a buzzer, FP is a condition when the predicted result is a buzzer and the actual label is non-buzzer, FN is a condition when the predicted result is a non-buzzer and the actual label is buzzer whereas TN is the condition when the predicted result is non-buzzer and the actual label is also non-buzzer.

3. Results and Discussions

In this experiment, we compare the performance of the three SVM functions using basic and full features. Feature analysis is also added using correlation values.

3.1 Classification Results using Basic Features

The experimental results on buzzer account classification using basic features as the baseline from this research with the F1-score performance measure can be seen in Table 6..

Tabel 6. F1 Score in Hashtagh Buzzer Classification using SVM and Basic Features (baseline)

		SVM Function		
		SVC	LSVC	NuSVC
K-Fold	1	0.42	**0.65**	0.60
	2	0.45	0.47	**0.71**
	3	0.33	0.40	**0.65**
	4	0.44	**0.70**	0.54
	5	0.44	**0.55**	0.48
	6	0.44	0.34	**0.60**
	7	0.44	**0.58**	0.54
	8	**0.49**	0.37	0.45
	9	0.49	0.56	**0.63**
	10	0.33	0.41	**0.80**
Average		0.42	0.50	0.60

Based on the experimental results in Table 6, the use of basic features in hashtag buzzer detection using various SVM models gives poor results. The highest F1 value obtained is 60% using the NSVC model.

Some other interesting findings are about the effectiveness of the SVC function in python in classifying this hashtag buzzer. The NuSVC function has an average value of F1 compared to the other two functions, although when viewed from the performance of each fold, NuSVC has a performance that is still comparable to LSVC. In addition, LSVC, which is a faster implementation of SVC, actually has a better F1 performance than SVC in almost every fold.

3.2 Classification Results using Complete Features

Complete Features are the basic features as shown in Table 3 plus the results of the expansion of the basic features as shown in Table 4. The results of the hashtag buzzer classification using complete features with the F1-score performance measure can be seen in Table 7.

Based on the experimental results in Table 7, it can be concluded that feature expansion using statistical value-based feature expansion is very effective in improving

the F1 performance of the SVM model in detecting hashtag buzzers. All SVM models used provide the same F1 performance, which is an average of 84%. In some values of k, the performance obtained reaches 100%. This value is more increased than the use of the basic features of account properties which only have an average F1 performance value ranging from 42% -60%.

Tabel 7. F1 Score in Hashtagh Buzzer Classification using SVM and Complete Features

		SVM Function SVC- LSVC- NuSVC
K-Fold	1	0.67
	2	1.00
	3	0.90
	4	0.85
	5	0.75
	6	0.75
	7	0.80
	8	0.75
	9	0.95
	10	1.00
Average		0.84

3.3 Feature Analysis

To further analyze the effect of the account property expansion features used, we also use another performance measure, namely the correlation value (r^2) to see how far the contribution of the proposed new features to the overall features used. The correlation value has a range of values [0,1]. The correlation value of feature X is 1, which means that feature X affects the classification result of a hashtag as a hashtag buzzer or not. The results of the calculation of the correlation of 10 features with the highest correlation value can be seen in Table 8.

Tabel 8. The 10 most influential features based on their correlation values

No	Feature	r^2
1	unique_user_pertweet	0.26
2	**Rg_F6**	0.21
3	**Rg_F1**	0.20
4	**Q2_F1**	0.20
5	unique_rt_pertweet	0.19
6	**Q3_F9**	0.19
7	url_pertweet	0.19
8	**Q2_F7**	0.16
9	**Q2_F1**	0.16
10	rt_pertweet	0.15

The results of the correlation analysis in Table 8 show that 60% of the features that have the highest correlation are new features resulting from the feature expansion process on account properties. This result is also in line with the results of the previous experiment in Table 7, that the addition of new features has been proven to be

able to improve the performance of the SVM model in detecting hashtag buzzers between 24-42%.

4. Conclusion

Based on the results of the experiments and analysis carried out, several conclusions were obtained, namely, the feature expansion based on descriptive statistics carried out in this study was proven to be effective in improving the performance of the SVM model in detecting the hashtag buzzer compared to the use of basic features (baseline). The combination of features and SVM classifier also proved effective in detecting hashtags buzzer. Another interesting thing obtained in this research is related to the implementation of SVM in the python language. The selection of the NuSVC function gives better results than the SVC and LinearSVC functions, especially when the features used are not very complete.

The future work of this research can be done by simulating all the SVM parameters completely to see the effect of the parameters on the performance of this SVM classifier. In addition, this research was also can be developed for different domains, for example in detecting hoax news or detecting buzzer accounts for other social media such as Instagram or Tiktok which are currently also very popular.

Reference

[1] Simon Kemp, "Digital 2021: Indonesia," 2021. https://datareportal.com/reports/digital-2021-indonesia.

[2] goodnewsfromindonesia, "Pengguna Facebook Indonesia dalam Bingkai Statistik," 2021. https://www.goodnewsfromindonesia.id/2021/10/30/penggun a-facebook-indonesia-dalam-bingkai-statistik (accessed Feb. 08, 2022).

[3] S. Preston, A. Anderson, D. J. R. Id, M. P. Shephard, and N. Huhe, "Detecting fake news on Facebook : The role of emotional intelligence," pp. 1–13, 2021, doi: 10.1371/journal.pone.0246757.

[4] A. J. Panatra, F. B. Chandra, W. Darmawan, H. L. H. S. Warnars, W. H. Utomo, and T. Matsuo, "Buzzer Detection to Maintain Information Neutrality in 2019 Indonesia Presidential Election," Proc. - 2019 8th Int. Congr. Adv. Appl. Informatics, IIAI-AAI 2019, pp. 873–876, 2019, doi: 10.1109/IIAI-AAI.2019.00177.

[5] E. Ferrara, "Disinformation and social bot operations in the run up to the 2017 French presidential election," First Monday, vol. 22, no. 8, 2017, doi: 10.5210/fm.v22i8.8005.

[6] H. Allcott and M. Gentzkow, "Trends in the Diffusion of Misinformation on Social Media," SAGE J. Res. Polit. Polit., vol. 6, no. 2, pp. 1–13, 2018.

[7] R. Mustika, "Pergeseran Peran Buzzer Ke Dunia Politik Di Media Sosial," Diakom J. Media dan Komun., vol. 2, no. 2, pp. 144–151, 2019, doi: 10.17933/diakom.v2i2.60.

[8] B. Arianto, "Salah Kaprah Ihwal Buzzer: Analisis Percakapan Warganet di Media Sosial," J. Ilm. Ilmu Pemerintah., vol. 5, no. 1, pp. 1–20, 2020, doi: 10.14710/jiip.v5i1.7287.

[9] S. Sugiono, "Fenomena Industri Buzzer Di Indonesia: Sebuah Kajian Ekonomi Politik Media," Commun. J. Ilmu Komun., vol. 4, no. 1, pp. 47–66, 2020, doi: 10.15575/cjik.v4i1.7250.

[10] A. Suciati, A. Wibisono, and P. Mursanto, "Twitter Buzzer Detection for Indonesian Presidential Election," ICICOS 2019 - 3rd Int. Conf. Informatics Comput. Sci. Accel. Informatics

Comput. Res. Smarter Soc. Era Ind. 4.0, Proc., 2019, doi: 10.1109/ICICoS48119.2019.8982529.

[11] S. Bradshaw and P. N. Howard, "The Global Disinformation Order 2019 Global Inventory of Organised Social Media Manipulation," Univ. Oxford, p. 25, 2019.

[12] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, "Detecting automation of Twitter accounts: Are you a human, bot, or cyborg?," *IEEE Trans. Dependable Secur. Comput.*, vol. 9, no. 6, pp. 811–824, 2012, doi: 10.1109/TDSC.2012.75.

[13] M. Ibrahim, O. Abdillah, A. F. Wicaksono, and M. Adriani, "Buzzer Detection and Sentiment Analysis for Predicting Presidential Election Results in a Twitter Nation," Proc. - 15th IEEE Int. Conf. Data Min. Work. ICDMW 2015, pp. 1348–1353, 2016, doi: 10.1109/ICDMW.2015.113.

[14] M. T. Juzar and S. Akbar, "Buzzer Detection on Twitter Using Modified Eigenvector Centrality," Proc. 2018 5th Int. Conf. Data Softw. Eng. ICoDSE 2018, pp. 1–5, 2018, doi: 10.1109/ICODSE.2018.8705788.

[15] C. Juditha, "Buzzer di Media Sosial Pada Pilkada dan Pemilu Indonesia Buzzer in Social Media in Local Elections and Indonesian Elections," pp. 199–212, 2019.

[16] D. Kotsakos, P. Sakkos, I. Katakis, and D. Gunopulos, "#tag: Meme or event?," ASONAM 2014 - Proc. 2014 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Min., no. Asonam, pp. 391–394, 2014, doi: 10.1109/ASONAM.2014.6921615.

[17] A. M. F. Al Sbou, A. Hussein, B. Talal, and R. A. Rashid, "A Survey of Arabic Text Classification Models," vol. 8, no. 6, pp. 4352–4355, 2018, doi: 10.11591/ijece.v8i6.pp.4352-4355.

[18] M. R. A. Utomo and Y. Sibaroni, "Text classification of british english and American english using support vector machine," 2019, doi: 10.1109/ICoICT.2019.8835256.

[19] Y. Sibaroni, D. H. Widyantoro, and M. L. Khodra, "Automatic identification of compare paper relations," *Int. J. Electr. Eng. Informatics*, vol. 12, no. 1, pp. 141–154, 2020, doi: 10.15676/ijeei.2020.12.1.12.

[20] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, "Hyperparameter Tuning for Machine Learning Algorithms Used for Arabic Sentiment Analysis," . Informatics, vol. 8, no. 79, pp. 1–21, 2021.

[21] P. Probst, Anne-Laure, and Bernd Bischl, "Tunability : Importance of Hyperparameters of Machine Learning Algorithms," J. Mach. Learn. Res., vol. 20, pp. 1–32, 2019.

[22] R. Hossain and D. Timmer, "Machine Learning Model Optimization with Hyper Parameter Tuning Approach," Glob. J. Comput. Sci. Technol. D Neural Artif. Intell., vol. 21, no. 2, 2021.

[23] S. Wang, L. Dong, and H. Hua, "Parameter optimization of support vector machine based on improved grid algorithm Parameter optimization of support vector machine based on improved grid algorithm," J. Phys. Conf. Ser., 2020, doi: 10.1088/1742-6596/1693/1/012108.

[24] L. C. Padierna and A. Rojas-dominguez, "Hyper-Parameter Tuning for Support Vector Machines by Estimation of Hyper-Parameter Tuning for Support Vector Machines by Estimation of Distribution Algorithms," no. December, 2017, doi: 10.1007/978-3-319-47054-2.

[25] M. A. Nanda and A. Maddu, "A Comparison Study of Kernel Functions in the Support Vector Machine and Its Application for Termite Detection," doi: 10.3390/info9010005.

[26] E. Pusporani, Suhartono, and D. D. Prastyo, "Hybrid multivariate generalized space-time autoregressive artificial neural network models to forecast air pollution data at Surabaya," AIP Conf. Proc., vol. 2194, 2019, doi: 10.1063/1.5139822.

[27] A. Karatzoglou, D. Meyer, and K. Hornik, "Journal of Statistical Software," vol. 15, no. 9, 2006.

[28] H. Lee, "ODE-based Epidemic Network Simulation of Viral Hepatitis A and Kernel Support Vector Machine based Vaccination Effect Analysis," J. Korean Inst. Intell. Syst., no. April, 2020, doi: 10.5391/JKIIS.2020.30.2.106.