ISSN 2278-3091

Volume 9, No.4, July – August 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse211942020.pdf

https://doi.org/10.30534/ijatcse/2020/211942020

Converting Text to Numerical RepresentationusingModified Bayesian Vectorization Technique for Multi-Class Classification

Hajah T. Sueno[1], Bobby D. Gerardo[2], Ruji P. Medina[3]

[1]Technological Institute of the Philippines-Quezon City, Philippines, qhsueno@tip.edu.ph

[2]West Visayas State University, Philippines, bgerardo@wvsu.edu.ph

[3]Technological Institute of the Philippines-Quezon City, Philippines, ruji.medina@tip.edu.ph

ABSTRACT

The first step towards making the text documents machine-readable is vectorization. Vectorisation allows the machines to understand textual content by transforming it into meaningful numerical representations. This study proposes a modified Bayesian vectorization and employing the Laplace smoothing method to reduce the dimensionality of features and improve the classification accuracy. Dataset of news articles was used in building the model and was evaluated across the metrics of precision, recall, F1-score, and accuracy. To validate the effectiveness of the enhancement, the model was compared to the Term Frequency and Inverse Document Frequency (TF-IDF) method. The results revealed that the proposed enhancement has significantly better results having 98% classification accuracy compared to 81% classification accuracy of the TF-IDF vectorization technique.

Key words: Modifiedvectorization, Document classification, Bayesian Vectorization, Support Vector Machine, Vectorization

1. INTRODUCTION

Machine learning features are numerical attributes that allow anyone to perform any mathematical operation from.Yet there are various cases where numerical attributes are not present in the dataset.These types of text data can not be fed directly into the machine for extracting features, because most algorithms expect the text's feature vectors as input [1][2]. And a way to translate each document into a numeric vector is required when dealing with text documents.This method is known as text vectorisation[3]. When the documents are converted into vectors, some kind of mathematical operation can be performed to use them for different purposes in text mining such as clustering, classification, ranking, summarization, etc. [4].

Most machine learning techniques employed in the area of text classification require the features of the documents to be effectively selected [5]. In conjunction with text classification, it has one limitation. This limitation involves the high dimensionality of feature space due to a large number of features. This occurs when the size of the text increases, the redundancy or noise of the data will increase as well [6]. Many dimensionality reduction methods were proposed in the last several years to solve this problem[5]. Vectorization techniques such as TF-IDF [7] vectorize the data easily, however, dimensionality is an issue since each vectorized word is used to represent the document. This results in the number of dimensions to be equal to the number of words. In this situation, one way to reduce dimensionality is to create a new set of features from the original feature set by extracting the features [8]. One of the first steps that were taken to solve this problem was to find a way to vectorize words.

This paper proposes a modified Bayesian vectorization model with the Laplace smoothing method to alleviate the problem of dimensionality and to improve classification accuracy. The enhancement is carried out to transform each of the text documents in the dataset into the format of probability distribution in the vector space using the Bayesian vectorizer then feed this probability distribution to Support Vector Machine (SVM) for training and classification purposes. To validate the improvements afforded by using Bayesian vectorization, results are compared to Term Frequency and Inverse Document Frequency (TF-IDF) method for Support Vector Machine (SVM) classifier in terms of Precision, Recall, F1-score, and accuracy.

The rest of the paper is arranged as follows. Section 2 discusses the literature review of feature vectorization, Bayesian vectorization, and smoothing method. Section 3 includes the methodology. Section 4 discusses the simulation results and discussion while Section 5 highlights the conclusion and recommendation.

2. LITERATURE REVIEW

2.1 Feature Vectorization

Feature vectorization is the process of transforming text data into a machine-understandable format to attain a more concise textual representation. This technique aims at building a new collection of features after applying a few transformations to the corpus.Typically because of the transformations the extracted features do not bear the same information as before[2].Nonetheless, it is likely to achieve more condensed data providing a great dimension reduction. In addition, the extracted features may reveal a better spatial representation of the data [9]. Figure 1 shows the feature vectorization process for machine learning.

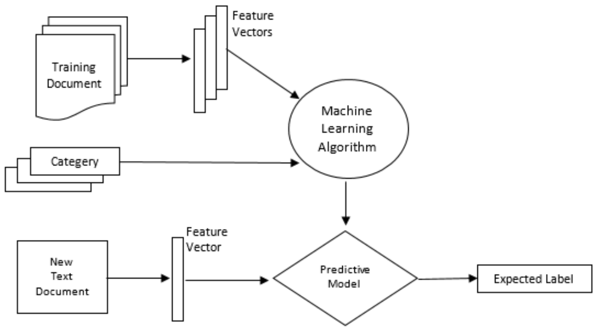


Figure 1: Feature vectorization for Machine Learning

2.2 Bayesian Vectorization

Machine learning algorithms most often take vectors of numerical features as input. And a way to translate each document into a numeric vector is required when dealing with text documents. This process is known as vectorization of text [2][10].

Bayesian vectorization for text documents used the distribution of probability, where the dimensions of the features are dependent on the number of categories available in the classification task[11]. It utilizes the raw text document for training purposes and the classifier utilizes the vectorized training data[12]. Bayesian vectorization relies on the Bayes formula with presumed independence between predictors, that used a set of training data to compute the posterior probability, which calculates the probability and estimates the probability conditions necessary for classification[13][14][14]. In the context of document classification, the Bayesian vectorization uses the probability that a specific document is annotated in a particular category, as the document has those terms in it, the likelihood of finding those particular words in that category, the probability that any document is annotated in that category, separated by the chances of obtaining those words in any document [15][16], as shown in equation (1):

$$Pr\ (Category|Word) = \frac{Pr\ (Word|Category) * Pr\ (Category)}{Pr\ (Word)} \quad (1)$$

Some studies have shown the usefulness of the use of Bayes theorem in different areas, such as safety [17][18], agriculture [19][20], image processing [21][22], etc.

Meneses et al. [23]proposed two basic solutions to the approximate formula of Bayes, when making precise decisions.The output was measured where a decision is made on one of two outcomes is most likely to occur and where a choice is made between an option that provides appropriate utility for something that is assured or a possibility that results in either a worse or a better benefit. Bayes theoremwasstudied[24]for an efficient and secure intelligent navigation method in classifying thousands of Navigational Talex messages gathered in navigational area VI. Based on the result, the optimal classifier's accuracy rate reaches 97 per cent. One of the research paper closely related was of[2][25][26], that used a Bayesian vectorization technique to preprocess the text documents and reduce the dimensions. The findings have improved classification performance.

2.3 Laplace Smoothing Method

Laplace smoothing is one of the smoothing method used in Naïve Bayes [27]. Not only do Laplace smoothing methods usually avoid zero probability, but they also try to improve the model's accuracy as a whole. [28], [29]. In a big dataset, the random choice of training data set in a large dataset would result in a zero value in the probability model. These zero values would make the input data classification impossible for the classifier. In this case, the smoothing method is used to eliminate zero value in the probability model. The implementation can be seen in equation (2) below. The equation is capable of handling zero probabilities by adding one (+ 1) to all terms.

$$P(tk|c) = \frac{count(tk|c) + 1}{count(t|c)|v + 1|} \quad (2)$$

P(tk|c): Likelihood probability, term probability for a class
count(tk|c): is the number of terms observed in a class
count(t|c): is the sum of all terms in a class
v: is the number of vocabulary in the training set

Previous studies show that there are differences in performance obtained by various smoothing techniques against various types of data. Laplace smoothing method [2]has provided better results in sentiment analysis, where 10-fold cross-validation is used on training data to obtain the optimal parameter [30]. Also studied were smoothing methods in conducting real-time stream data that investigated a scheme implementing Laplace smoothing technique with Binarized Naïve Bayes Classifier (NBC) to improve accuracy and using SparkR to speed up.[31].

Laplace's performance was better than Dirichlet and Absolute Discounting. However, the difference in performance results can be seen in other researches [32]–[34] where Laplace smoothing achieved lower performance when compared to other smoothing techniques. The difference in performance smoothing technique can occur due to factors of noise word features in the training data. Noise word features are words that do not strongly represent the characteristic of sentiment class resulting in lower performance in classification [30], [34], [35]

3. METHODOLOGY

In this study, a modified Bayesian vectorization model is developed to reduce the dimensionality of data and to produce a higher classification accuracy. Figure 2 shows the proposed modified Bayesian vectorization model.

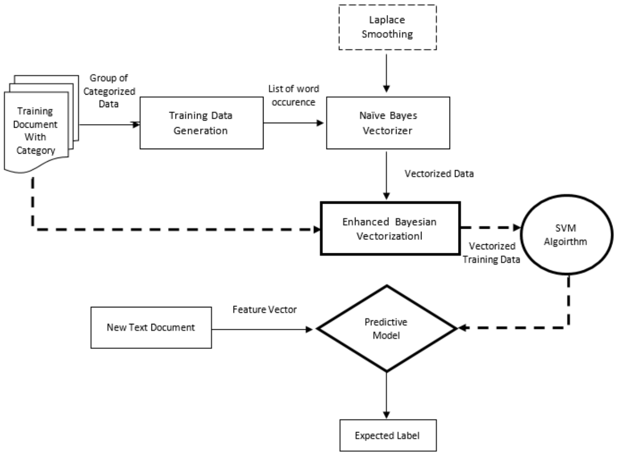


Figure 2: ModifiedBayesian Vectorization Model

3.1Data Source

In this study, the 2012 to 2018 news headlines obtained from HuffPost, which was originally compiled by Misra[36] is used. Every news headline was manually labelled according to the following 5 categories using a spreadsheet namely – urban, science, arts, politics, and travel with a total of 2640 documents. Dataset had been split into a training set with 1848 documents and a testing set with 792 documents.

3.2 Preprocessing Methods

Standardization of words is performed at this point to remove accentuation, punctuation, special characters, and numbers. All letters are converted into lowercase letters, with each character replaced by single space except alphabets and multiple spaces. Noisy data such as headers of text files, footers, HTML , XML, and markup data are excluded.

3.3 Building the Improved Naïve Bayes Vectorization

To extract each word from document (X) a simple word extraction algorithm is used to produce a list of words containing the number of occurrences of each word in category (C). The same approach is used in the Training data set to produce the number of all words in each group.

The prior probability of each group can then be determined by means of equation (3):

$$Pr\ (C) = \frac{\text{Total Number of Document in Category}}{\text{Total\_Number\_Of\_Document\_in\_Training\_Dataset}} \qquad (3)$$

The equation (4) below is used to measure the probability of a given category for a particular word.

$$Pr\ (X|\ C) = \frac{\text{Occurrence of Word In the Category}}{\text{Total\_Number\_of\_All\_Words\_in\_Category}} \qquad (4)$$

Laplace smoothing is applied to prevent a situation of zero probability and to ensure that each word has a chance of occurrence, based on at least one count, even if it does not appear in the training results. The count is increased to a small value(usually 1)[2][37][38] using the equation (5):

$$Pr\ (X|\ C) = \frac{\text{Occurrence of Word In the Category } +1}{\text{Total\_Number\_of\_All\_Words\_in\_Category} + |V| +1}$$

where |V| is the total unique words in the training set (5)

The overall probability of an annotation of a text to a given category is determined using the equation (6):

$$Pr\ (C|\ X) = Pr\ (W_1|\ C) * Pr\ (W_2|C) * Pr\ (W_3|C) * Pr\ (W_n|C) * Pr\ (C) \qquad (6)$$

The mathematical *log*function is applied using the equation to prevent this underflow error (7):

$$Pr\ (C\ |\ X) = log\ (Pr\ (W_1|C) + log\ (Pr\ (W_2|C) + log\ (Pr\ (W_3|C) + log\ (Pr\ (W_n|C) + log\ (Pr\ (C)) \qquad (7)$$

The logarithmic function is applied since log increases or decreases monotonically which means the order of probabilities will not be affected. Smaller probabilities will remain smaller once the log is applied to them, and vice versa. This can effectively avoid the can pitfall of an underflow error without affecting the predictions of the trained model. [39].

3.4Evaluating the Model

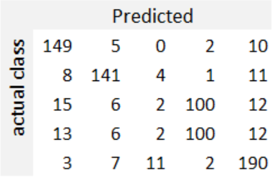
Machine learning toolkit LIBSVM[40]was used to train the classification models. The model was trained using a collection of well-categorized vectorized training data given by the vectorizer at Naïve Bayes. The vectorized training data was divided by running the SVM into a 70 percent training set (1,848 documents) and a 30 percent test set (792 documents) for the classification.In the training data set, a 10-fold cross-validation was performed to determine the robustness of the predictions from the SVM models. The remainder of the classification tasks are performed using the linear kernel function with parameter C being set to 1.

4. SIMULATION RESULTS AND DISCUSSIONS

Simulation is performed to compare the SVM classification result using the TF-IDF vectorization technique and the modified Bayesian vectorization model with a smoothing method to determine whether the proposed modified Bayesian vectorization model results in better classification accuracy as compared to the TF-IDF vectorization in SVM classifier.

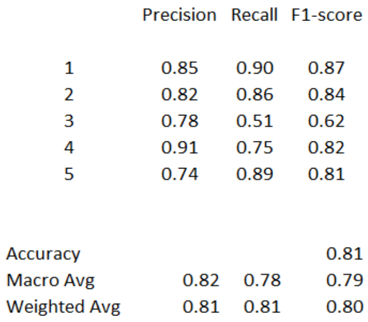
4.1Simulation Result using TF-IDF Technique with SVM Classifier

TF-IDF-SVM classification correctly classified 149 urban news, 141 science news, 2 art news, 100 politics news, and 190 travel news as shown in Figure 3. It achieved an average Precision value of 82%, Recall value 78%, F1-score 79%, and Accuracy of 81% as shown in Figure 4.



	Predicted				
actual class	149	5	0	2	10
	8	141	4	1	11
	15	6	2	100	12
	13	6	2	100	12
	3	7	11	2	190

Figure 3: TF-IDF-SVM Confusion Matrix



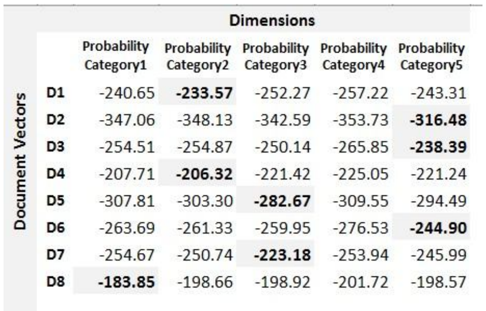
	Precision	Recall	F1-score
1	0.85	0.90	0.87
2	0.82	0.86	0.84
3	0.78	0.51	0.62
4	0.91	0.75	0.82
5	0.74	0.89	0.81
Accuracy			0.81
Macro Avg	0.82	0.78	0.79
Weighted Avg	0.81	0.81	0.80

Figure 4: TF-IDF-SVM Classification Result

4.2 Simulation Result using ModifiedBayesian Technique with SVM Classifier

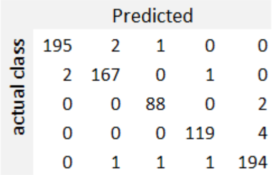
Figure 5 contains the generated vectorized training data set using the modified Bayesian vectorization model.Each dimension in a simplified sense represents a meaning, and the numerical weight of the document on that dimension captures the closeness of its association with and to that meaning. The value highlighted in a document represents the highest probability that implies the dimension or category that it belongs to.

As shown in Figure 6, the Bayesian-SVM classification correctly classified 195 urban news, 167 science news, 88 arts news, 119 politics news, and 194 travel news.



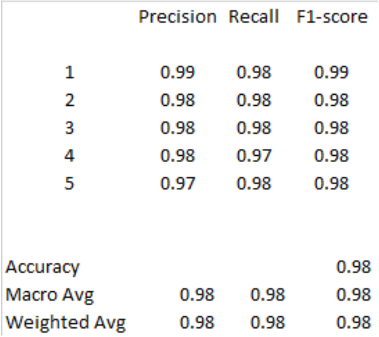
		Dimensions				
		Probability Category1	Probability Category2	Probability Category3	Probability Category4	Probability Category5
Document Vectors	D1	-240.65	**-233.57**	-252.27	-257.22	-243.31
	D2	-347.06	-348.13	-342.59	-353.73	**-316.48**
	D3	-254.51	-254.87	-250.14	-265.85	**-238.39**
	D4	-207.71	**-206.32**	-221.42	-225.05	-221.24
	D5	-307.81	-303.30	**-282.67**	-309.55	-294.49
	D6	-263.69	-261.33	-259.95	-276.53	**-244.90**
	D7	-254.67	-250.74	**-223.18**	-253.94	-245.99
	D8	**-183.85**	-198.66	-198.92	-201.72	-198.57

Figure 5: Sample Vectorized Training Data



	Predicted				
actual class	195	2	1	0	0
	2	167	0	1	0
	0	0	88	0	2
	0	0	0	119	4
	0	1	1	1	194

Figure 6: Modified Bayesian-SVM Confusion Matrix



	Precision	Recall	F1-score
1	0.99	0.98	0.99
2	0.98	0.98	0.98
3	0.98	0.98	0.98
4	0.98	0.97	0.98
5	0.97	0.98	0.98
Accuracy			0.98
Macro Avg	0.98	0.98	0.98
Weighted Avg	0.98	0.98	0.98

Figure 7: Modified Bayesian-SVM Classification Result

The modified model achieved an average accuracy of 98% as shown in Figure 7, which is significantly higher than the TF-IDF-SVM classification approach. The modified model also gives the highest Precision, Recall, and F1-score values.

5. CONCLUSION

The modified Bayesian-SVM classifier outperforms the TF-IDF-SVM classifier, based on comparison using the same news headlines dataset.The proposed enhancement has been observed to be highly productive and to identify the documents with great precision. Laplace's use of smoothing to improve Bayesian-SVM has achieved a classification accuracy of 98 percent compared to the 81 percent classification accuracy of TF-IDF-SVM.These results showed that the Bayesian vectorization technique has contributed to the SVM classifier with a more effective textual data transformation process compared to the same purpose using the TFIDF vectorization technique. Future works might also investigate the effects of other smoothing techniques.

ACKNOWLEDGEMENT

Wewould like to thank Technological Institute of the Philippines, Quezon City (TIP-QC) for giving us useful comments and suggestion and CHED K12 Program for the scholarship program.

REFERENCES

[1] D. Saxena, S. K., and K. N., “Survey Paper on Feature Extraction Methods in Text Categorization,” *Int. J. Comput. Appl.*, vol. 166, no. 11, pp. 11–17, 2017 doi: 10.5120/ijca2017914145.

[2] H. T. Sueno, B. D. Gerardo, and R. P. Medina, “Multi-class document classification using support

vector machine (SVM) based on improved naïve bayes vectorization technique," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 3, pp. 3937–3944, 2020, doi: 10.30534/ijatcse/2020/216932020.

[3] B. Bengfort, R. Bilbro, and T. Ojeda, *Applied Text Analysis with Python: Enabling Language-Aware Data Products with Machine Learning*, 1st ed. O'Reilly Media Incorporated, 2018.

[4] R. K. Roul, J. K. Sahoo, and K. Arora, "Modified TF-IDF Term Weighting Strategies for Text Categorization," *2017 14th IEEE India Counc. Int. Conf. INDICON 2017*, pp. 0–5, 2018, doi: 10.1109/INDICON.2017.8487593.

[5] T. Sabbah, M. Ayyash, and M. Ashraf, "Hybrid Support Vector Machine based Feature Selection Method for Text Classification," vol. 15, no. 3, pp. 599–609, 2018.

[6] A. K. Uysal and S. Gunal, "A novel probabilistic feature selection method for text classification," *Knowledge-Based Syst.*, vol. 36, pp. 226–235, 2012, doi: 10.1016/j.knosys.2012.06.005.

[7] P. Soucy and G. W. Mineau, "Beyond TFIDF weighting for text categorization in the vector space model," *IJCAI Int. Jt. Conf. Artif. Intell.*, pp. 1130–1135, 2005.

[8] G. . Ramadevi and K. Usharani, "Study on Dimensionality Reduction Techniques and Applications," *Publ. Probl. Appl. Eng. Res.*, vol. 04, no. 1, pp. 134–140, 2013.

[9] C. H. P. Ferreira, F. O. De Franca, and D. R. Medeiros, "Combining Multiple Views from a Distance Based Feature Extraction for Text Classification," *2018 IEEE Congr. Evol. Comput. CEC 2018 - Proc.*, 2018, doi: 10.1109/CEC.2018.8477772.

[10] M. Syamala and N.J.Nalini, "A Deep Analysis on Aspect based Sentiment Text Classification Approaches," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 4, no. 2, pp. 15–21, 2019, doi: https://doi.org/10.30534/ijatcse/2019/01852019.

[11] D. Isa, L. H. Lee, V. P. Kallimani, and R. Rajkumar, "Text document preprocessing with the bayes formula for classification using the support vector machine," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 9, pp. 1264–1272, 2008, doi: 10.1109/TKDE.2008.76.

[12] L. H. Lee, C. H. Wan, R. Rajkumar, and D. Isa, "An enhanced Support Vector Machine classification framework by using Euclidean distance function for text document categorization," *Appl. Intell.*, vol. 37, no. 1, pp. 80–99, 2012, doi: 10.1007/s10489-011-0314-z.

[13] O. J. Okesola, K. O. Okokpujie, A. A. Adewale, S. N. John, and O. Omoruyi, "An Improved Bank Credit Scoring Model: A Naïve Bayesian Approach," *Proc. - 2017 Int. Conf. Comput. Sci. Comput. Intell. CSCI 2017*, pp. 228–233, 2018, doi: 10.1109/CSCI.2017.36.

[14] J. Rexiline Ragini and P. M. Rubesh Anand, "An empirical analysis and classification of crisis related tweets," *2016 IEEE Int. Conf. Comput. Intell. Comput. Res. ICCIC 2016*, pp. 2–5, 2017, doi: 10.1109/ICCIC.2016.7919608.

[15] D. Buzic and J. Dobsa, "Lyrics classification using Naive Bayes - IEEE Conference Publication," *2018 41st Int. Conv. Inf. Commun. Technol. Electron. Microelectron.*, p. 5, 2018, doi: 10.3390/genes6020372.

[16] M. Baygin, "Classification of Text Documents based on Naive Bayes using N-Gram Features," *2018 Int. Conf. Artif. Intell. Data Process. IDAP 2018*, pp. 1–5, 2019, doi: 10.1109/IDAP.2018.8620853.

[17] D. Leman, "Expert System Diagnose Tuberculosis Using Bayes Theorem Method and Shafer Dempster Method," *2018 6th Int. Conf. Cyber IT Serv. Manag. CITSM 2018*, no. Citsm, pp. 1–4, 2019, doi: 10.1109/CITSM.2018.8674380.

[18] N. Ramkumar, S. Prakash, S. A. Kumar, and K. Sangeetha, "Prediction of liver cancer using Conditional probability Bayes theorem," *2017 Int. Conf. Comput. Commun. Informatics, ICCCI 2017*, pp. 7–11, 2017, doi: 10.1109/ICCCI.2017.8117752.

[19] F. Ikorasaki and M. B. Akbar, "Detecting Corn Plant Disease with Expert System Using Bayes Theorem Method," *2018 6th Int. Conf. Cyber IT Serv. Manag. CITSM 2018*, no. Citsm, pp. 1–3, 2019, doi: 10.1109/CITSM.2018.8674303.

[20] C. Cassandra and R. Sari, "Agricultural Expert System Design Based on Bayes Theorem," *Proc. 2018 Int. Conf. Inf. Manag. Technol. ICIMTech 2018*, no. September, pp. 315–320, 2018, doi: 10.1109/ICIMTech.2018.8528127.

[21] X. Deng, J. Guo, Y. Chen, and X. Liu, "A method for detecting document orientation by using Naïve Bayes classifier," *Proc. 2012 Int. Conf. Ind. Control Electron. Eng. ICICEE 2012*, pp. 429–432, 2012, doi: 10.1109/ICICEE.2012.120.

[22] L. R. A. X. Menezes and A. J. F. Loureiro, "CmWave through vegetation: Correlation of pixels and attenuation using UT and Bayes Inference," *2017 IEEE Antennas Propag. Soc. Int. Symp. Proc.*, vol. 2017-Janua, pp. 1829–1830, 2017, doi: 10.1109/APUSNCURSINRSM.2017.8072957.

[23] P. Goodwin, "When simple alternatives to Bayes formula work well: Reducing the cognitive load when updating probability forecasts," *J. Bus. Res.*, vol. 68, no. 8, pp. 1686–1691, 2015, doi: 10.1016/j.jbusres.2015.03.027.

[24] H. Liu, Z. Liu, X. Wang, and Y. Cai, "Bayes' Theorem based maritime safety information classifier," *Proc. 30th Chinese Control Decis. Conf. CCDC 2018*, pp. 2725–2729, 2018, doi: 10.1109/CCDC.2018.8407588.

[25] L. H. Lee, C. H. Wan, R. Rajkumar, and D. Isa, "An enhanced Support Vector Machine classification framework by using Euclidean distance function for text document categorization," *Appl. Intell.*, vol. 37, no. 1, pp. 80–99, 2012, doi: 10.1007/s10489-011-0314-z.

[26] L. H. Lee, R. Rajkumar, and D. Isa, "Automatic folder allocation system using Bayesian-support vector

machines hybrid classification approach," *Appl. Intell.*, vol. 36, no. 2, pp. 295–307, 2012, doi: 10.1007/s10489-010-0261-0.

[27] A. F. Cahyanti, R. Saptono, and S. widya Sihwi, "Determination of the Best Model in the Naive Bayes Classifier Method in Determining Toddler Nutrition Status by Considering Independence of Parameters," *J. ITSMART*, vol. 4, no. 1, 2015.

[28] S. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," *Comput. Speech Lang.*, vol. 13, no. 4, pp. 359–393, 1999.

[29] K. P. Murphy, "Naive Bayes Classifier," 2006.

[30] R. A. Ramadhani, F. Indriani, and D. T. Nugrahadi, "Comparison of Naive Bayes smoothing methods for Twitter sentiment analysis," *2016 Int. Conf. Adv. Comput. Sci. Inf. Syst. ICACSIS 2016*, pp. 287–292, 2017, doi: 10.1109/ICACSIS.2016.7872720.

[31] Y. G. Jung, K. T. Kim, B. Lee, and H. Y. Youn, "Enhanced Naive Bayes Classifier for real-Time sentiment analysis with SparkR," *2016 Int. Conf. Inf. Commun. Technol. Converg. ICTC 2016*, pp. 141–146, 2016, doi: 10.1109/ICTC.2016.7763455.

[32] Q. Yuan, G. Cong, and N. M. Thalmann, "Enhancing Naive Bayes with various smoothing methods for short text classification," in *WWW'12 - Proceedings of the 21st Annual Conference on World Wide Web Companion*, 2012, pp. 645–646, doi: 10.1145/2187980.2188169.

[33] M. Hafilizara and J. Adisantoso, "Metode Smoothing dalam Naive Bayes untuk Klasifikasi Email Spam," in *Bogor: Institut Pertanian Bogor. (in Bahasa Indonesia)*, 2014.

[34] M. Kikuchi, M. Yoshida, M. Okabe, and K. Umemura, "Confidence interval of probability estimator of Laplace smoothing," *ICAICTA 2015 - 2015 Int. Conf. Adv. Informatics Concepts, Theory Appl.*, 2015, doi: 10.1109/ICAICTA.2015.7335387.

[35] R. Rismanto, D. W. Wibowo, and A. R. Syulistyo, "Implementation of Naive Bayes Classifier and Log Probabilistic for Book Classification Based on the Title," *Int. J. Eng. Technol.*, vol. 4, pp. 131–136, 2018, doi: 10.14419/ijet.v7i4.36.28987.

[36] R. Misra, "News Headlines Dataset For Sarcasm Detection," doi: 10.13140/RG.2.2.16182.40004.

[37] S. Aggarwal and D. Kaur, "Enhanced Smoothing Methods Using Naïve Bayes Classifier for Better Spam Classification," vol. 2, no. 9, pp. 3061–3073, 2013.

[38] N. Sharma and M. Singh, "Modifying Naive Bayes classifier for multinomial text classification," *2016 Int. Conf. Recent Adv. Innov. Eng. ICRAIE 2016*, 2017, doi: 10.1109/ICRAIE.2016.7939519.

[39] A. Javed, "Unfolding Naïve Bayes from Scratch," *Towards Data Science*, 2018. https://towardsdatascience.com/unfolding-naïve-bayes-from-scratch-2e86dcae4b01 (accessed Aug. 12, 2019).

[40] C. C. Chang and C. J. Lin, "LIBSVM: A Library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–39, 2011 doi: 10.1145/1961189.1961199.