BAREKENG: Journal of Mathematics and Its Applications
June 2024 Volume 18 Issue 2 Page 1297–1306
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol18iss2pp1297-1306

TEXT CLASSIFICATION USING ADAPTIVE BOOSTING ALGORITHM WITH OPTIMIZATION OF PARAMETERS TUNING ON CABLE NEWS NETWORK (CNN) ARTICLES

Dewi Retno Sari Saputro[1*], **Krisna Sidiq**[2], **Harun Al Rasyid**[3], **Sutanto**[4]

[1,2,3,4] *Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Sebelas Maret*
Jln. Ir. Sutami No. 36, Surakarta, 57126, Indonesia

*Corresponding author's e-mail: * **dewiretnoss@staff.uns.ac.id***

ABSTRACT

Article History:

Received: 14th January 2024
Revised: 25th March 2024
Accepted: 3rd May 2024
Pubished: 1st June 2024

Keywords:

AdaBoost;
Classification;
News Article;
Parameter Tuning.

The development of the era encourages advances in communication and information technology. This resulted in the exchange of information being faster because it is connected to the internet. One platform that provides online news articles is Cabel News Network (CNN), which has been broadcasting news on its website since 1995. The number of Cabel News Network news articles continues to increase, so news articles are categorized to make it easier for readers to find articles according to the category they want. Classification is a technique for determining the class of an object based on its characteristics, where the class label is known beforehand. One of the algorithms for classification is adaptive boosting (AdaBoost). The AdaBoost algorithm performs classification by building several weighted decision trees (stumps), and then the class determination is based on the number of stumps with the highest weight. The AdaBoost algorithm can be combined with parameter tuning to avoid overfitting or underfitting resulting from a weak set of stumps. Therefore, this study implements the AdaBoost algorithm with parameter tuning on CNN news article classification. The data used in this study is CNN news article data from 2011 to 2022 sourced from the Kaggle page. The data is categorized into six classes, namely business, entertainment, health, news, politics, and sports. This study uses two evaluation metrics, namely the accuracy value and the confusion matrix, to measure the performance of the AdaBoost algorithm. The accuracy value obtained is 0,78763, the precision value is 0.91, the recall value is 0.85, and the F1 score value is 0.88

How to cite this article:

D. R. S. Saputro, K. Sidiq, H. A. Rasyid and Sutanto., "TEXT CLASSIFICATION USING ADAPTIVE BOOSTING ALGORITHM WITH OPTIMIZATION OF PARAMETERS TUNING ON CABLE NEWS NETWORK (CNN) ARTICLES," *BAREKENG: J. Math. & App.*, vol. 18, iss. 2, pp. 1297-1306, June, 2024.

Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id*

Research Article · Open Access

1. INTRODUCTION

The development of the era encourages advances in information and communication technology. One of the influences of technological developments is the change in an individual in obtaining information. Communication technology has changed the journalism system that previously existed in mass media, such as newspapers, magazines, television, and radio, into an internet-based digital platform [1]. The growth of the internet encourages the exchange of digital information to occur in a short time in large amounts. The information can be in the form of text, such as news articles.

A news article is a journalistic work written based on facts or event data [2]. Various digital platforms have provided news articles that can be accessed via internet-connected gadgets, one of which is the Cable News Network (CNN). CNN is a news channel from the United States that broadcasts news through web media, blogs, television, and radio. CNN launched a website in 1995 and began broadcasting news articles online [3]. The news articles continue to grow every year and have a large number. A large number of news articles needs to be managed in order to facilitate access to certain news article topics. News articles belong to the data in the form of the text so data mining is used to process them.

Data mining is a process of finding hidden patterns and new information from large database warehouses to assist in decision-making or knowledge discovery [4]. Furthermore, the various techniques contained in data mining have different functions and results, namely association, clustering, estimation, prediction, generalization, visualization, and classification [5]. Classification is a data mining technique that functions to determine the class of a given object based on its characteristics [6]. The classification is classified as supervised learning because the data used has been labeled class. Classification is divided into two types based on the class, namely binary and multi-class. One of the supervised learning algorithms that can perform multi-class classification is the adaptive boosting algorithm (AdaBoost).

The advantage of the AdaBoost algorithm is that it corrects decision-making errors in the next iteration so that it can improve predictor accuracy [7]. In addition, the AdaBoost algorithm also utilizes bagging and boosting techniques in building several decision trees to obtain predictive data. The AdaBoost algorithm needs to be combined with parameter tuning to avoid overfitting or underfitting as a result of a weak set of individual decision trees [8]. Therefore, this study implemented the AdaBoost algorithm with parameter tuning in the classification of CNN news articles.

2. RESEARCH METHODS

This research was conducted in five stages, namely research data, text preprocessing, classification using the AdaBoost algorithm, parameter tuning with search cross-validation, and evaluation.

2.1 Research Data

Data is a group of raw information or facts which can be in the form of symbols, numbers, words, or images. One of which is often found in the form of words, namely online news articles. This research uses CNN news article data from 2011 to 2022 sourced from the Kaggle page. A lot of news article data is used, namely 37.949 data. This study uses four variables, namely index, category, headline, and article text, which was developed from Arimbawa and Sanjaya's research, which only uses three variables, namely article title, article content, and article label [9]. The index variable is needed to indicate the position of data in the table even though the order of the data changes when performing data preprocessing. The category variable contains the news article classification class, the headline variable contains the news article title, and the article text variable contains the news article text document.

2.2 Text Preprocessing

Text preprocessing is a process for selecting text data to make it more structured through a series of stages. One of the text preprocessing methods is natural language processing (NLP). NLP is a branch of artificial intelligence that deals with interactions between computers and humans using natural language. NLP is tasked with calculating sentiment and determining which part of a language is considered important [10]. The initial stage of NLP is text preprocessing, which is tasked with preparing unstructured text data into structured data that is ready for processing [11]. In preparing data, text preprocessing performs various stages. There are no rules that regulate by default regarding the stages and sequences in text preprocessing because everything depends on the type and condition of the data used. In general, the stages in text preprocessing are described as follows [12].

2.2.1 Parsing

Parsing is the process of breaking/decomposing a document into several parts. A document generally consists of several parts, so it needs to be described to take the important parts needed.

2.2.2 Case Folding

The text data obtained is generally inconsistent with capital letters. Therefore, case folding is responsible for generalizing the use of capital letters in a document. In addition, characters other than letters and numbers, such as spaces and punctuation marks, are considered delimiters that can be deleted or ignored in the document.

2.2.3 Tokenizing

Tokenizing is the stage of cutting text or sentences into small parts called tokens. The token serves to facilitate data analysis because the data is in the form of word fragments.

2.2.4 Filtering

Filtering is in charge of selecting words that are considered important from the results of tokenizing. Meanwhile, words that are considered unimportant and have no meaning (stopword) will be removed from the document. The goal is to reduce the size of the index so that data processing time becomes shorter.

2.2.5 Stemming

The stemming stage is the process of changing the form of a word into its basic word by removing the suffix or prefix. This stage is important because it can minimize the number of different indexes in one data and group words that have similar meanings.

2.2.6 Vectorization

Text vectorization is a technique for converting text into a vector with values in the form of the number of occurrences of terms (unique words). Vectorization is important so that the text is easily understood by machines. One of the vectorization methods is Term Frequency-Inverse Document Frequency (TF-IDF). TF is the frequency of occurrence of term t in document d. Meanwhile, IDF functions to reduce the weight of a term if its appearance is widely spread throughout the document. The term t frequency (TF) and term t weight (IDF) are formulated in Equation (1) and Equation (2).

$$TF = \begin{cases} 1 + \log_{10}(f_{t,d}), f_{t,d} > 0 \\ 0, f_{t,d} = 0 \end{cases} \tag{1}$$

$$IDF_t = \log_{10}\left(\frac{D}{df_t}\right) \tag{2}$$

with

$f_{t,d}$: frequency of term t in documents d,

D : the sum of all documents in the collection, and

df_t : the number of documents containing the term t.

2.3 Classification Using the AdaBoost Algorithm

Classification is a data mining technique that functions to determine the class of a given object based on its characteristics [6]. One classification algorithm is AdaBoost which is built from a collection of several forest stumps. Stumps forest is a simple decision tree made up of one branch and two leaves. Each stumps affects the final prediction because it has a different weight; the greater the error the stumps have, the smaller the weight or influence on the final prediction [7]. The prediction results of the AdaBoost algorithm are obtained by selecting the largest weight from the sum of each stump with the same prediction class as illustrated in Figure 1.

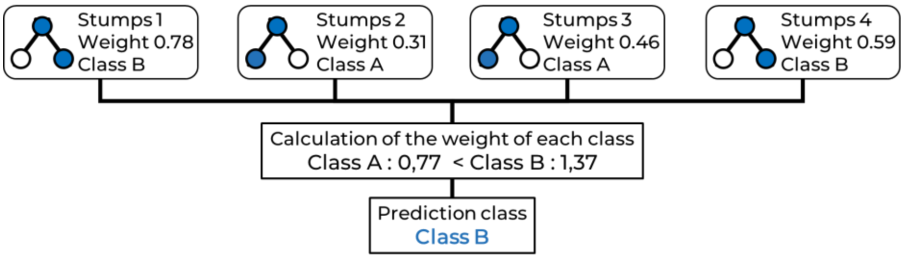

Figure 1. AdaBoost Algorithm Architecture

Based on Figure 1, there are four stumps, the first and fourth of which are classified as class B, with a weight of 1.37. Meanwhile, the second and third stumps are classified as class A with a weight of 0.77. Thus, classification is based on the largest weight, namely class B. The determination of the weight of each stump is based on the error rate resulting from the algorithm training. In the algorithm training process, each row of data will be weighted based on the classification errors made. The following is the initialization weight for each x-th row of data in the AdaBoost algorithm [8].

$$D_0(x) = \frac{1}{N} \tag{3}$$

with

$D_0(x)$: the weight of the x-th data row in the d-th document, and
N : the total number of data rows.

After determining the weights, the next step is to iterate (for $t = 1, 2, \ldots, T$) in building a collection of stumps which are described as follows.

1. Determine the weakest classifier $h_t(x)$ using the weights $D_t(x)$
2. Calculate the error rate ϵ_t

$$\epsilon_t = \sum_{i=1}^{N} D_t(x_i).I[y_i \neq h_t(x_i)] \tag{4}$$

3. Assign α_t weights to the $h_t(x)$ classifier

$$\alpha_t = \log\left(\frac{1-\epsilon_t}{\epsilon_t}\right) \tag{5}$$

4. For each x_i

$$D_t(x_i) = D_t(x_i).\exp(\alpha_t.I[y_i \neq h_t(x_i)]) \tag{6}$$

5. Normalize $D_t(x_i)$ so that

$$\sum_{i=1}^{N} D_t(x_i) = 1 \tag{7}$$

6. Output

$$H(x) = sign[\sum \alpha_t h_t(x)] \tag{8}$$

with

t : iteration step,
$h_t(x)$: the weakest classifier in the x-th data row,
$D_t(x)$: weight of the x-th data row in t iteration,
ϵ_t : error rate,
y_i : training sample i-th,
α_t : stumps weigh, and
$H(x)$: final classifier hypothesis on the x-th data row.

2.4 Parameter Tuning with Search Cross-Validation

Parameter tuning is the process of adjusting parameters in a classification model to improve its performance. The AdaBoost algorithm is able to work efficiently if the stumps that are built are able to understand the dataset well so that it has a small error. Therefore, the AdaBoost algorithm requires setting the right combination of parameters in the dataset to avoid overfitting or underfitting cases. One method that can be used to determine the best parameter combination is Search Cross-Validation (SearchCV) [13]. SearchCV is a method for selecting combinations of parameters and models by testing each combination one by one and validating each combination to produce the best model performance.

2.5 Evaluation

Evaluation is the stage of measuring the performance of a model that has been made so that it can be considered for choosing the best model. The evaluation metrics used in the classification case are the accuracy value, precision, recall, and F1 score. The accuracy value is obtained by dividing the number of correct predictions by the total number of predictions, which is shown below [14].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{9}$$

with

TP (True Positive) : the detected positive data is correct,
TN (True Negative) : negative data detected is true,
FP (False Positive) : negative data but detected as positive data, and
FN (False Negative) : positive data but detected as negative data.

Precision is a metric that gives the proportion of true positives to the total number of positives that the model predicts.

$$Precision = \frac{TP}{TP+FP} \tag{10}$$

Recall focuses on how good the model is at finding all the positives.

$$Recall = \frac{TP}{TP+FN} \tag{11}$$

As can be seen from the definitions of precision and recall, they are closely related. The F1 score is a measure that combines recall and precision. As has been seen, there is a trade-off between precision and recall. Therefore, F1 can be used to measure how effectively the model makes that trade-off.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{12}$$

All evaluation metrics are obtained from the confusion matrix. The confusion matrix displays and compares the actual values with the predicted values in the classification case [15].

3. RESULTS AND DISCUSSION

3.1 Data Preprocessing

Before CNN news article data is classified, the data needs to be cleaned and processed first. CNN news articles consist of nine categories, namely news, sports, politics, business, health, entertainment, travel, VR, and style. However, the travel, VR, and style categories have a very small amount of data that interferes with the prediction results. Therefore, it is necessary to eliminate and balance the data. The next stage is case folding, namely the elimination of characters or words that are considered unimportant (stopwords). In this study, all characters or symbols were removed from the document, and the numeric characters were converted to words so that the document contained only letter characters. The purpose of deleting these characters is to facilitate and speed up data processing. The list of words that are considered unimportant (stopwords) is adapted to the stopwords package taken from the Python NLTK (Natural Language Tool Kit) library, such as the words are, a, the, is, or, and so on, totaling 179 words.

The next stage is the lemmatization process to group words with the same base word and stemming to remove the suffix or prefix in the word. The purpose of this stage is to reduce the document index so that the document only contains the main words, and the data processing process becomes shorter. The next stage is dividing the data into two, namely training data and test data, with proportions of 85% and 15%. The data-sharing process uses the train_test_split package taken from the Sklearn Python library with random_state 8. The last stage in data preprocessing is vectorization. The method used in this study is TF-IDF, which is also taken from the Sklearn Python library. Vectorization aims to give weight to each word in the document. This study vectorizes 1000 words. The higher the weight of a word, the higher the frequency of occurrence of the word. Therefore, in this study the classification is based on the frequency of occurrence of a word in the document. For example, documents in the sports category have a high frequency of words related to sports, and the health category has a high frequency of words related to health.

3.2 Data Classification with AdaBoost Algorithm

The AdaBoost algorithm used in this study is taken from the Sklearn Python library with random_state 2 and default parameters. The AdaBoost algorithm performs classification by building a set of decision trees (stumps). In the formation of each stump, the prediction error will be accumulated as an error rate (ϵ). In this study, 50 stumps were built and an error rate was generated for each stump. The error rate is influenced by two things, namely, the number of features (word vectorization) used and the selection of the model. In this research, the more features used, the lower the error rate will be.

The next step is to calculate the weight (α) of each stump based on the resulting error rate (ϵ). Based on **Equation (5)**, with the natural logarithm (e), if the error rate is $\epsilon > 0{,}5$ the weight value is $\alpha < 0$ and vice versa. The weight of each stump is summed according to the same classification class, then the final result is based on the classification class with the largest weight. In this study, the accuracy value is used to measure the performance of the algorithm. Based on **Equation (9)** an accuracy value of 0.70845 is obtained for the training data and 0.69086 for the testing data so that the algorithm is not good enough to classify.

3.3 Parameter Tuning with Grid Search Cross-Validation

Grid Search Cross-Validation is a method for determining the best parameter combination taken from the Sklearn Python library. In this study, the combined parameters are n_estimators, learning_rate, and boosting, for random_state set to a value of 2. The definition of n_estimators is the number of stumps (decision trees) built, and learning_rate is the value of the weight correction during training. The boosting parameters in the AdaBoost algorithm are divided into two, namely Stagewise Additive Modeling using a Multiclass Exponential loss function (SAMME) and Stagewise Additive Modeling using a Real Multiclass Exponential loss function (SAMME.R). The more parameters and values that are set in it, the longer the Grid Search Cross-Validation process will take. **Figure 2** shows a comparison of the accuracy of the algorithms based on predetermined parameter values.

Based on **Figure 2**, the n_estimators parameter has a large influence on the accuracy value. The more stumps formed, the higher the accuracy value. In addition, the learning_rate parameter shows high accuracy when the learning_rate value is close to 1. The learning_rate < 1 parameter has good classification results when combined with SAMME.R boosting. Based on **Figure 2** (e), learning_rate = 1 has good classification results with a combination of SAMME boosting. SAMME.R uses probability estimation to update additive models, whereas SAMME only uses classifications to update additive models. **Table 1** shows the highest accuracy values for each algorithm training with different learning_rate.

Table 1. Highest Accuracy for Any Learning_Rate

Learning_rate	Highest accuracy	N_estimators	Boosting
0.0001	0.37366	1000	SAMME.R
0.001	0.65323	1000	SAMME.R
0.01	0.75269	1000	SAMME.R
0.1	0.75806	900	SAMME
1	0.78763	1000	SAMME

Based on **Table 1**, the fifth row of the second column, the algorithm with the highest accuracy has the parameter values learning_rate = 1, n_estimators = 1000, and boosting SAMME. In the classification process without Grid Search Cross-Validation, the algorithm produces an accuracy value of 0.69086. In contrast, the classification process with Grid Search Cross-Validation produces a higher accuracy value of 0.78763. Thus, the Grid Search Cross-Validation method can improve the accuracy of the AdaBoost algorithm in classifying CNN news articles.

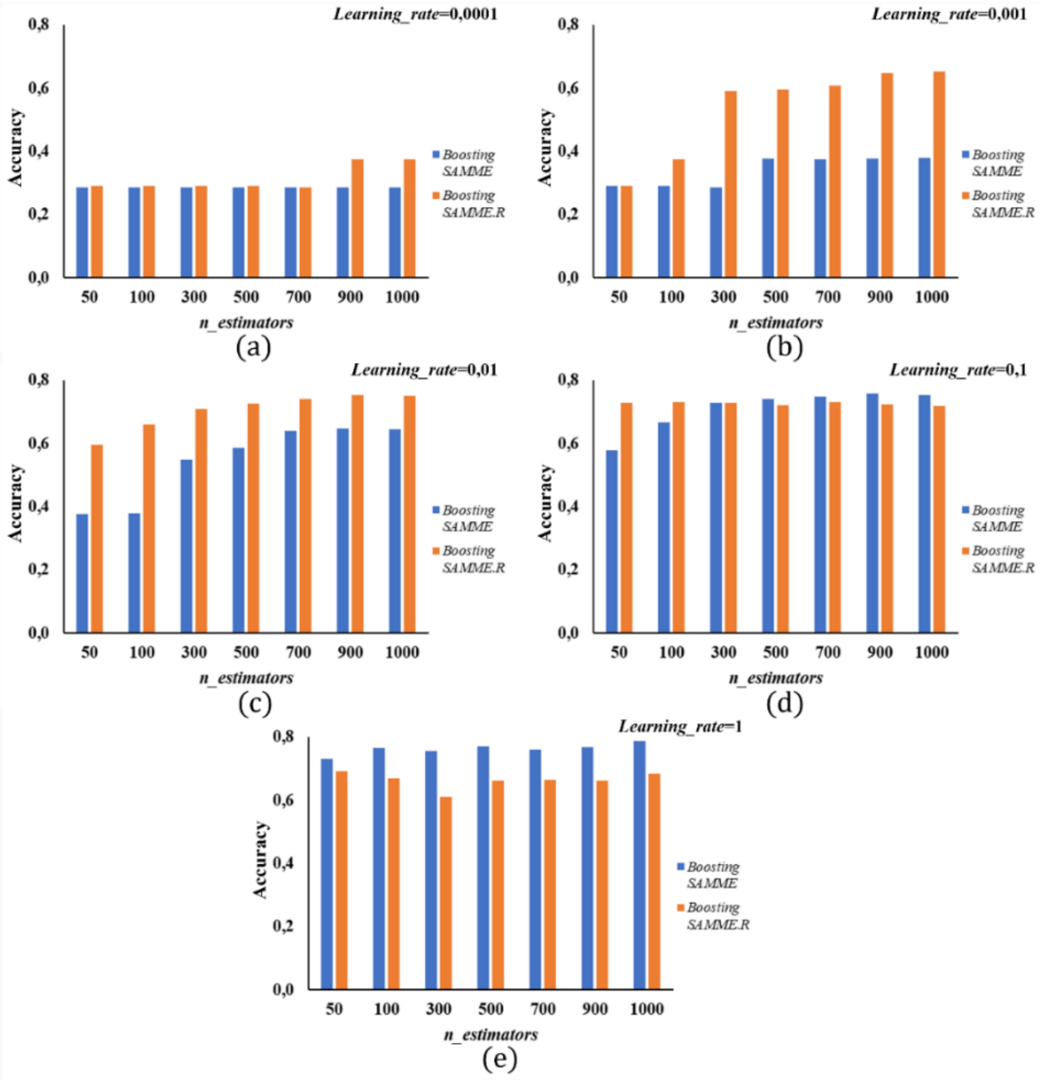

Figure 2. Bar Chart of Comparison of Accuracy Values Based on Boosting and Learning_Rate Parameters (a) 0,0001, (b) 0,001, (c) 0,01, (d) 0,1, and (e) 1.

3.4 Evaluation of Classification Results

The classification results obtained are then evaluated using the accuracy value evaluation metric and the confusion matrix. Based on **Equation (9)** an accuracy value of 0.81853 is obtained for the training data and 0.78763 for the testing data so that the algorithm used avoids overfitting or underfitting. Furthermore, the confusion matrix shows the comparison between the actual data (y) and the predicted data (y') where the columns represent the actual data and the rows represent the predicted data. Each cell in the confusion matrix shows the amount of data predicted in each class so that prediction accuracy and prediction errors can be known. **Figure 3** is the confusion matrix resulting from the AdaBoost algorithm classification.

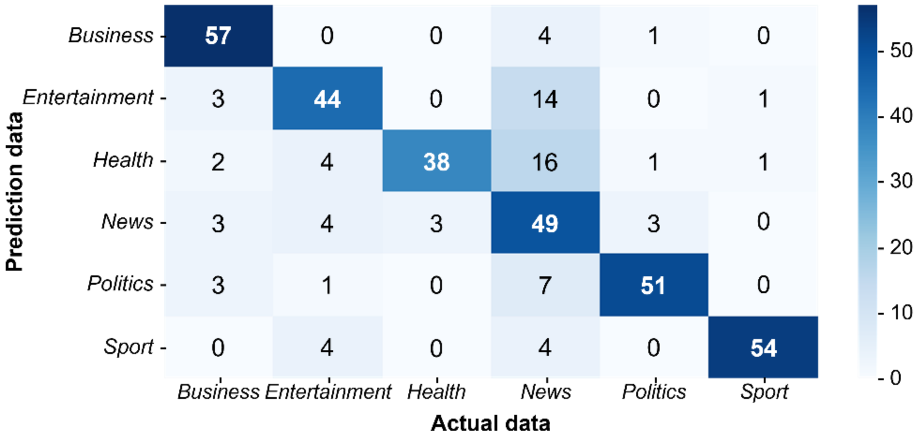

Figure 3. Confusion Matrix of Classification Results

Based on Figure 3, the business category was predicted to be correct with 57 data and 5 data incorrectly predicted, the entertainment category was predicted to be correct with 44 data and 18 data incorrectly predicted, the health category was predicted to be correct with 38 data and incorrectly predicted with 24 data, the news category was predicted correctly with 49 data and incorrectly predicted with 13 data, the politics category was predicted to be correct with 51 data and 11 data incorrectly predicted, and the sports category was predicted to be correct with 54 data and incorrect predictions with 8 data. The evaluation results show that most of the CNN news articles can be classified correctly using the AdaBoost algorithm.

Based on Figure 3 and Equations (10-12), the evaluation metrics for precision, recall, and F1 score can be calculated as follows. The precision value is 0.91, the recall value is 0.85, and the F1 score value is 0.88. This value shows quite good classification results for CNN news articles. Thus, the AdaBoost algorithm with tuning parameters is able to classify CNN news articles into six categories.

4. CONCLUSIONS

Based on the results and discussion, it can be concluded that the AdaBoost algorithm is able to classify CNN news articles into six categories, namely, business, entertainment, health, news, politics, and sport. The accuracy value of the classification results using the AdaBoost algorithm was initially 0.69086 then after optimization using parameter tuning, the accuracy value increased to 0.78763. In addition, the precision value obtained is 0.91, the recall value is 0.85, and the F1 score value is 0.88. Thus, parameter tuning can improve the performance of the classification algorithm quite significantly. Based on Figure 3, It can be concluded that the AdaBoost algorithm with parameter tuning is able to predict the majority of CNN news article categories correctly, so that these results can facilitate grouping large numbers of online news articles easily and efficiently.

REFERENCES

[1] S. A. Eldridge, K. Hess, E. C. Tandoc, and O. Westlund, “Navigating the Scholarly Terrain: Introducing the Digital Journalism Studies Compass,” *Digit. Journal.*, vol. 7, no. 3, pp. 386–403, 2019, doi: 10.1080/21670811.2019.1599724.

[2] C. Juditha, “Akurasi Berita dalam Jurnalisme Online (Kasus Dugaan Korupsi Mahkamah Konstitusi di Portal Berita Detiknews),” *J. Pekommas*, vol. 16, no. 3, pp. 145–154, 2013, [Online]. Available: https://media.neliti.com/media/publications/222363-akurasi-berita-dalam-jurnalisme-online-k.pdf.

[3] O. D. Ugo, O. C. Uzoma, Izuogu, and K. Chukwuemeka, “COMMUNICATION AUDIT OF CABLE NEWS NETWORK (CNN) ONLINE REPORTS ON BOKO HARAM INSURGENCY IN NIGERIA (2012-2016),” vol. 3, no. 5, pp. 19–34, 2017, [Online]. Available: https://eajournals.org/ijirmmcs/vol-3-issue-5-october-2017/communication-audit-cable-news-network-cnn-online-reports-boko-haram-insurgency-nigeria-2012-2016/.

[4] M. A. Jassim and S. N. Abdulwahid, “Data Mining preparation: Process, Techniques and Major Issues in Data Analysis,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1090, no. 1, p. 012053, 2021, doi: 10.1088/1757-899x/1090/1/012053.

[5] S. H. Liao, P. H. Chu, and P. Y. Hsiao, “Data mining techniques and applications - A decade review from 2000 to 2011,” *Expert Syst. Appl.*, vol. 39, no. 12, pp. 11303–11311, 2012, doi: 10.1016/j.eswa.2012.02.063.

[6] A. O. Adebayo and M. S. Chaubey, “Data Mining Classification Techniques on the Analysis of Student’s Performance,” *Glob. Sci. J.*, vol. 7, no. 4x, pp. 79–95, 2019, doi: 10.11216/gsj.2019.04.19671.

[7] C. Tu, H. Liu, and B. Xu, “AdaBoost typical Algorithm and its application research,” *MATEC Web Conf.*, vol. 139, 2017, doi: 10.1051/matecconf/201713900222.

[8] R. Gao and Z. Liu, “An Improved AdaBoost Algorithm for Hyperparameter Optimization,” *J. Phys. Conf. Ser.*, vol. 1631, no. 1, 2020, doi: 10.1088/1742-6596/1631/1/012048.

[9] I. G. A. Purnajiwa Arimbawa and N. A. Sanjaya ER, “Penerapan Metode Adaboost Untuk Multi-Label Classification Pada Dokumen Teks,” *JELIKU (Jurnal Elektron. Ilmu Komput. Udayana)*, vol. 9, no. 1, p. 127, 2020, doi: 10.24843/jlk.2020.v09.i01.p13.

[10] S. R. Joseph, H. Hloman, K. Letsholo, and K. Sedimo, “Natural Language Processing: A Review,” *Int. J. Res. Eng. Appl. Sci.*, vol. 6, no. 3, pp. 1–8, 2016, [Online]. Available: http://www.euroasiapub.org.

[11] V. Gurusamy and S. Kannan, “Preprocessing Techniques for Text Mining,” *Int. J. Comput. Sci. Commun. Networks*, vol. 5, no. 1, pp. 7–16, 2014, [Online]. Available: https://www.researchgate.net/publication/273127322_Preprocessing_Techniques_for_Text_Mining.

[12] A. I. Kadhim, “An Evaluation of Preprocessing Techniques for Text Classification,” *Int. J. Comput. Sci. Inf. Secur.*, vol. 16, no. 6, pp. 22–32, 2018, [Online]. Available: https://sites.google.com/site/ijcsis/.

[13] D. Mesafint and M. D. Huchaiah, “Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results,” *Int. J. Comput. Appl.*, vol. 44, pp. 1–12, 2021, doi: 10.1080/1206212X.2021.1974663.

[14] Nofriani, “Comparations of Supervised Machine Learning Techniques in Predicting the Classification of the Household’s Welfare Status,” *J. Pekommas*, vol. 4, no. 1, pp. 43–52, 2019, doi: 10.30818/jpkm.2019.2040105.

[15] M. Hossin and M. N. Sulaiman, “A Review on Evaluation Metrics for Data Classification Evaluations,” *Int. J. Data Min. Knowl. Manag. Process*, vol. 5, no. 2, pp. 01–11, 2015, doi: 10.5121/ijdkp.2015.5201.