BERT for Sentiment Analysis on Rotten Tomatoes Reviews

Aji Gautama Putrada
Advanced and Creative Networks Research Center
Telkom University
Bandung, Indonesia
ajigps@telkomuniversity.ac.id

Nur Alamsyah
Advanced and Creative Networks Research Center
Telkom University
Bandung, Indonesia
nuralamsyah@student.telkomuniversity.ac.id

Mohamad Nurkamal Fauzan
Advanced and Creative Networks Research Center
Telkom University
Bandung, Indonesia
mnurkamalfauzan@student.telkomuniversity.ac.id

2023 International Conference on Data Science and Its Applications (ICoDSA) | 979-8-3503-0519-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICoDSA58501.2023.10276800

***Abstract*—Transformer is a neural network model whose trend is rising because ChatGPT shocked the world with its general answering and questioning system capabilities. The bidirectional encoder representation transformer (BERT) is also one of the other pre-trained transformer models. However, there are research opportunities to use BERT in sentiment analysis for the Rotten Tomatoes review. Our research aims to use BERT for sentiment analysis on the Rotten Tomatoes dataset and evaluate the results. Our research step is to get the Rotten Tomatoes dataset from the Huggingface dataset. We do pre-processing for the BERT model, tune the BERT model, then train the model using the dataset. We use the lightweight pre-trained DistilBERT as part of our proposed model. We compare the performance of our BERT model with three state-of-the-art benchmark methods: support vector machine with term frequency-inverse document frequency (SVM+TF-IDF), naïve Bayes with TF-IDF (NB+TF-IDF), and convolutional neural network (CNN). We use several test metrics such as accuracy, precision, recall, f1-score, and area under the curve (AUC) from the receiver operating curve (ROC). In the training optimization process, we learned that AdamW is more effective than Adam optimizer. Our test results show that BERT's performance is better than the other three models with Accuracy, Precision, Recall, and F1-Scores of 0.753, 0.754, 0.754, and 0.753, respectively. Finally, BERT is also the best in terms of AUC, with a value of 0.826.**

***Index Terms*—bidirectional encoder representation transformer, text classification, rotten tomatoes, sentiment analysis, distilBERT, adamW**

I. Introduction

Transformer is a neural network model whose trend is increasing [1]. ChatGPT, which has taken the world by storm because of its general answering and questioning system capabilities, is a form of pre-trained transformer called generative pre-trained transformer (GPT) [2]. Several fields of artificial intelligence (AI) for the application of transformers are natural language processing (NLP) and computer vision (CV) [3]. The Bidirectional encoder representation transformer (BERT) is one of the other pre-trained transformer models besides GPT [4].

As a pre-trained model, BERT is suitable for sentiment analysis through the transfer learning technique [5]. As research conducted by [6] succeeded in detecting hate speech from social media Twitter using the BERT approach through transfer learning techniques, the results show the best performance compared to other models. The BERT model with transfer learning techniques can also solve sentiment analysis problems with a high degree of difficulty, such as in Bengali, which has 160 different forms of verb inflections, achieving an accuracy level of 71% [7].

Rotten Tomatoes is a website that classifies film reviews by critics into several categories, including good, bad, or currently using sentiment analysis. Several studies have used several methods for sentiment analysis on Rotten Tomatoes reviews. For example, Soni *et al.* [8] used a support vector machine (SVM) plus term frequency-inverse document frequency (TF-IDF) and got the f1-score performance of 0.77. Frangidis *et al.* [9] used naïve Bayes (NB) with TF-IDF on the Rotten Tomatoes dataset and obtained an f1-score of 0.80. Sankar *et al.* [10] used convolutional neural network (CNN) on Rotten Tomatoes and the Internet Movie Database (IMDb) dataset and got the best f1-score of 0.65. Our research problem and summary for the research gap from previous research is to use BERT on sentiment analysis for Rotten Tomatoes reviews.

Our research aims to use BERT for sentiment analysis on the Rotten Tomatoes dataset and evaluate the results. Our research is limited to using the Rotten Tomatoes dataset from the Huggingface dataset. We do pre-processing for the BERT model, tune the BERT model, then train the model using the dataset. We compared the performance of our BERT model with three state-of-the-art benchmark methods: SVM+TF-IDF, NB+TF-IDF, and CNN. We employ a number of test metrics, including accuracy, precision, recall, f1-score, and area under the receiver operating curve (AUC) (ROC).

To our knowledge, no research has used BERT for sentiment analysis on Rotten Tomatoes reviews. Here are our research contributions:

- A small and lightweight use of DistilBERT for Rotten Tomatoes review sentiment analysis
- A DistilBERT model for sentiment analysis using AdamW optimizer where the performance is better than using Adam
- A BERT-based transformer model that outperforms SVM+TF-IDF, NB+TF-IDF, and CNN in sentiment anal-

We gratefully acknowledge the support for this study provided by Telkom University's research and community service (PPM) directorate.

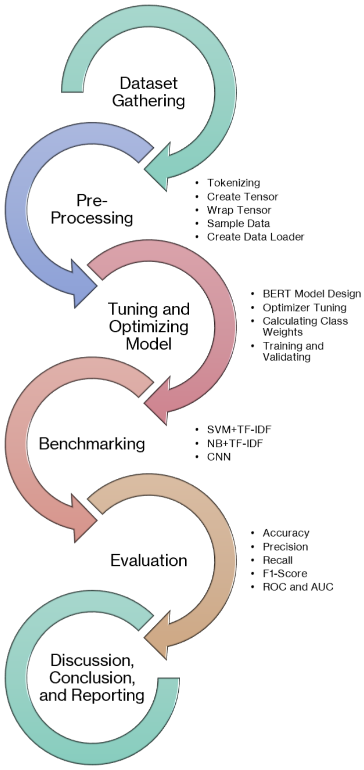

Fig. 1. The proposed methodology for the BERT Model.

ysis for Rotten Tomatoes reviews

The following systematics are employed for the rest of this essay: Section II describes our research method and the formulas we use. Section III displays the results of the tests we performed on the BERT model. Finally, Section IV contains the conclusions of our research.

II. Research Method

We propose a methodology for our BERT model. Our research step is to get the Rotten Tomatoes dataset from the Huggingface dataset. We do pre-processing for the BERT model, tune the BERT model, then train the model using the dataset. We compare the performance of our BERT model with four state-of-the-art benchmark methods: SVM+TF-IDF, NB+TF-IDF, and CNN. We use several test metrics such as accuracy, precision, recall, f1-score, and AUC from the ROC. Fig. 1 is a workflow diagram that shows our methodology.

A. The Rotten Tomatoes Dataset and Text Pre-Processing

We obtained our Rotten Tomatoes dataset from Huggingface datasets [11]. The dataset has two features: $text(string)$ and $label(classlabel)$. The $text(string)$ feature contains a Rotten Tomatoes review. Then, there are two data labels: 0 for negative sentiment and 1 for positive sentiment [12]. We took 700 data items from the training dataset and 300 from the testing datasets for 1000 data items. There are 491 data with negative labels and 509 with positive labels. There are no NaN data in the dataset. Table I shows some of the data items in the dataset. In the table, we provide three examples of Rotten Tomatoes reviews with negative labels, then two examples of Rotten Tomatoes Reviews with positive labels.

TABLE I
DATA ITEMS EXAMPLE IN THE HUGGINGFACE ROTTEN TOMATOES DATASET

text	label
"a modest and messy metaphysical thriller offering more questions than answers ."	0
"well , it does go on forever ."	0
"the film's tone and pacing are off almost from the get-go ."	0
"polished , well-structured film ."	1
"if your taste runs to 'difficult' films you absolutely can't miss it ."	1

We use PyTorch for our BERT platform. PyTorch is a machine learning framework developed for computer vision and natural language processing. PyTorch has two advantages: a library for deep learning and Tensors supported by the graphical processing unit (GPU) [13]. We go through several stages in pre-processing before the data can be used for the training process. These stages are as follows:

1) Tokenizing: Splits the sentence in the string into an array of [14] words. The result of tokenizing is token ID, with attention mask [15].
2) Create Tensor: Tensors are multidimensional matrices that contain various elements for training and testing purposes [16]
3) Wrap Tensor: Wrap Tensor combines ID tokens, attention masks, and labels into a single data structure
4) Sample Data: Sampling data for the training process
5) Create Data Loader: The Data Loader creates a data structure suitable for training iterations. Its other functions are random sampling, batch control, multiprocessing with the GPU, and memory pinning [17].

B. BERT for Sentiment Analysis

We use DistilBERT as our pre-trained transformer model [18]. DistilBERT is a BERT that goes through a distillation process, making the model smaller and lighter [19]. We created a transfer learning model using DistilBERT as part of its model [20]. Our model has an input size = 20. Then enter the pre-trained DistilBERT layer. Next are two neural network layers with sizes of 768 and 512 layers, respectively. These layers use the ReLU activation and dropout layers with probability = 0.5. The output layer is a neural network layer with 256 nodes and softmax activation with size 2. Fig. 2 shows the proposed BERT architecture for Rotten Tomatoes review sentiment analysis.

Furthermore, we use several parameters for the training process. We use 160 epochs, where the number results from empirical experiments. Then we use AdamW as the learning optimizer. AdamW is an optimization method where the weight is separated from the gradient update and calculated again when the weight updates [21]. The weight formula (θ_t) updates with AdamW to be as in (1).

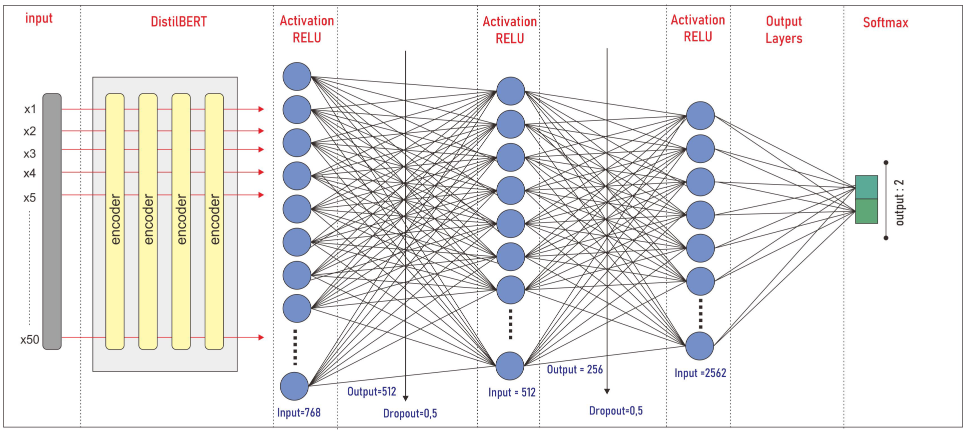

Fig. 2. The proposed BERT model architecture for Rotten Tomatoes review sentiment analysis.

$$\theta_t = \theta_{t-1} - \eta_t(\frac{\alpha \hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} + \lambda \theta_{t-1}) \tag{1}$$

where η stands for the scheduler factor, α for learning rate, $\hat{m}_t$ stands for moving average of gradient, $\hat{v}_t$ stands for moving average of gradient's square, $epsilon$ for error, and λ for weight decay.

Moreover, we use a learning rate of 10^{-5}. Then We use the cross-entropy loss for learning curve analysis. Cross entropy loss results from calculating the expected value between the predicted value distribution and the actual value distribution [22]. The cross-entropy loss formula ($H(y', y)$) is as in (2).

$$H(y, y') = -\sum_{n \in N} y_n \log y'_n \tag{2}$$

where y is the test labels, y' is the predicted labels, and N is the amount of data. Next, we divide our data, namely 70% for training, then 30% for validation. We use the learning rate scheduler to optimize the learning rate value whilst training [23]. We use a linear learning rate scheduler. Table II shows a summary of our settings for BERT training.

C. Benchmark Models and Performance Metrics

We benchmark our proposed model with three state-of-the-art models: SVM+TF-IDF [8], NB+TF-IDF [9], and CNN [10]. We use the TF-IDF for the SVM and NB models as a feature extraction method for text datasets [24]. The TF function gives weight to a word based on its appearance in a document. The higher the TF value, the more a word appears in a document. Meanwhile, IDF functions to give weight to a word based on the appearance of the word in all documents in a corpus. The greater the IDF value, the fewer documents have that word in a corpus. The final TF-IDF score is the multiplication of the TF and IDF scores.

TABLE II
OPTIMIZED PARAMETERS FOR BERT TRAINING

Parameter	Value
Epochs	160
Optimizer	AdamW
Learning Rate	10^{-5}
Learning Curve Metrics	Cross Entropy Loss
Train Data : Validation Data	70%:30%
Learning Rate Scheduler	Linear

SVM and NB are two methods that use TF-IDF. SVM is a method that classifies two labels based on the linear distance between the two labels [25]. Suppose the two labels cannot be separated linearly. In that case, SVM can perform a kernel trick, namely transforming the data into a higher dimension so that the transformed data can be separated linearly [26]. Various kernel types exist, including sigmoid, polynomial, and radial basis functions (RBF) [27]. Here we use RBF, where the advantage of RBF is to separate data with the concept of distance, similar to the k-nearest neighbor (KNN) [28]. Then NB is a method that classifies based on Bayes theorem [29]. NB is a classification method that often excels in text analysis case studies such as sentiment analysis [30]. Algorithm 1 shows the training process for the SVM+TF-IDF and NB+TF-IDF models, including their pre-processing stage. The $Remove\_Stopwords(Text)$ process removes simple words like "and," "or," and "if." Then there are two word-simplification treatments, namely stemming and lemmatizing. We chose to lemmatize over stemming, because lemmatizing returns a word to its root form, instead of stemming, which removes affixes from a word. The process is represented by $Lemmatize(Text)$.

CNN is a deep learning method that applies convolutional kernels throughout the dataset in its feature learning stage [31]. CNN is a well-known method for image processing, where 2D-CNN is used in image processing. Meanwhile, for time-series data, which is one-dimensional, there is 1D-CNN [32]. We use the Keras library for our 1D-CNN implementation. Table III shows our 1D-CNN architecture for Rotten Tomatoes review sentiment analysis. Our input is 100 in size because we embedded it with $max\_length = 100$. The first layer is the embedding layer which has a function like the normaliza-

Algorithm 1 The Process of SVM+TF-IDF and NB+TF-IDF

Require: $Text, Labels$
Ensure: y_{SVM}, y_{NB}
$Text \leftarrow Normalize(Text)$ ▷ Pre-Processing Stage
$Text \leftarrow Tokenize(Text)$
$Text \leftarrow Remove\_Stopwords(Text)$
$Text \leftarrow Lemmatize(Text)$

$X \leftarrow TF\_IDF(Text)$ ▷ TF-IDF Stage
$y \leftarrow Labels$

$SVM\_Model \leftarrow SVM\_Train(X, y)$ ▷ SVM Learning Stage
$y_{SVM} \leftarrow SVM\_Model(X)$

$NB\_Model \leftarrow NB\_Train(X, y)$ ▷ NB Learning Stage
$y_{NB} \leftarrow NB\_Model(X)$

tion function [32]. Then there is global max pooling which functions to generalize the results of CNN [33]. Rectified linear activation function (ReLU) gives output equal to input if positive and output equal to zero if negative [34]. Finally, the softmax activation function maps the categorical output to the logistic function curve [35].

We evaluate our models using accuracy, precision, recall, f1-score, ROC, and AUC. Accuracy compares all correct predictions to all predictions. Precision evaluates the correct prediction composition of all prediction results. Meanwhile, recall evaluates the composition of the correct predictions from all actual data. The harmonic average of recall and precision is known as the f1-score. As the decision-making threshold of a model changes, the relationship between the true positive rate (TPR) and the false positive rate (FPR) is explained by the ROC curve [36]. AUC is an objective interpretation of the ROC curve and stands for the area under the ROC curve [37]. Applying the trapezoidal rule to the ROC curve results in the calculation of AUC [38].

III. Results and Discussion

A. Results

We implement our proposed sentiment analysis model using DistilBERT. We compared two optimizers during training: AdamW and Adam. Each optimizer uses $learning\_rate = 10^{-5}$. Then we used 160 epochs for its training. Fig. 3 shows the loss curve of the BERT training process between AdamW and Adam. At the same time, Fig. 4 shows the loss curve from the BERT validation process between using AdamW and Adam. Qualitatively, the AdamW process is slightly better than Adam in the cross-entropy loss.

TABLE III
1D-CNN Architecture for Rotten Tomatoes Review Sentiment Analysis

Layer	Type	Shape	Activation
Input	Input	100	
Layer 1	Embedding	100	
Layer 2	1D-CNN	64 × 3	RelU
Layer 3	1D-CNN	64 × 3	RelU
Layer 4	1D-Global Max Pooling	64	
Layer 5	Dense	10	RelU
Output	Dense	2	Softmax

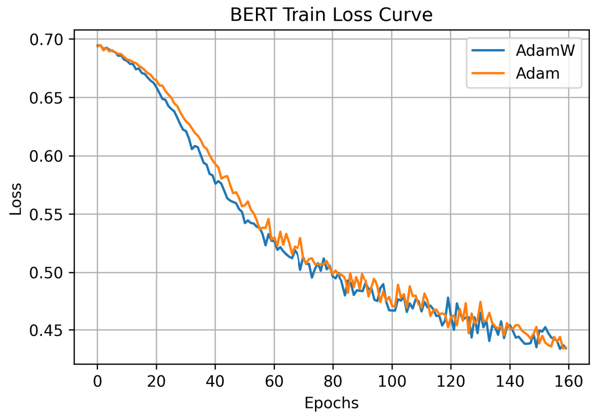

Fig. 3. The BERT training loss curve comparison between AdamW and Adam.

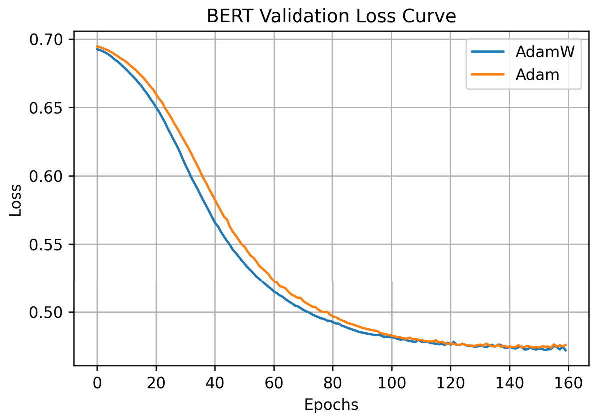

Fig. 4. The BERT validation loss curve comparison between AdamW and Adam.

We compare the performance of the DistilBERT model using the AdamW optimizer and Adam. Table IV compares the performance of the BERT model using the two optimizers. DistilBERT+AdamW has better TP than DistilBERT+Adam. DistilBERT+Adam has 0.42 times the FP of DistilBERT+AdamW. Nevertheless, DistilBERT+Adam's FN is higher than DistilBERT+AdamW. Further calculation results show that DistilBERT+AdamW's Accuracy and F1-Score are higher than DistilBERT+Adam, 0.74 and 0.75, respectively.

We implemented SVM+TF-IDF, NB+TF-IDF, and CNN so they can become benchmarks for BERT in sentiment analysis. To compare the four models, we first use accuracy, precision, recall, and f1-score. The performance of the four models is shown in Fig. 5, with values of 0.753, 0.754, 0.754, and 0.753 for accuracy, precision, recall, and f1-score, respectively; BERT outperforms the other three models. Furthermore,

TABLE IV
PERFORMANCE COMPARISON BETWEEN THE DISTILBERT+ADAMW MODEL AND THE DISTILBERT+ADAM MODEL

Model Name	TP	TN	FP	FN	Accuracy	Precision	Recall	F1-Score
DistilBERT+AdamW	58	53	19	20	0.74	0.75	0.74	0.75
DistilBERT+Adam	46	64	8	32	0.73	0.85	0.59	0.70

NB+TF-IDF is the model with the worst performance. CNN is the model with the second-best performance. We also tried several ensemble models: random forest and adaptive boosting (AdaBoost). The accuracy results are 0.632 for random forest and 0.604 for AdaBoost.

Next, we evaluate the ROC and AUC of the four models. Fig. 6 shows the ROC curves of the four models. Qualitatively, BERT has the best ROC. CNN, SVM+TF-IDF follow this model, and finally, NB+TF-IDF. BERT has the highest AUC value, which is 0.83. Meanwhile, in order of highest values, CNN, SVM+TF-IDF, and NB+TF-IDF have AUCs of 0.712, 0.702, and 0.665, respectively.

B. Discussion

We use DistilBERT, which has a pre-trained model size smaller than the original BERT [19]. We measure our model size, which is 255.16 MB. In contrast, the original BERT model we tried for the Rotten Tomatoes review sentiment analysis has a larger size, namely 419.16 MB. Our research contribution is a small and lightweight DistilBERT for Rotten Tomatoes review sentiment analysis.

A previous study used DistilBERT with Adam optimizer for drug re-purposing in COVID-19 cases [39]. Here we use AdamW and show that its performance results are better than Adam at the same learning rate and the number of epochs. Our research contribution is a DistilBERT model for sentiment analysis using AdamW optimizer, where the performance is better than using Adam.

Several other studies have offered a state-of-the-art model for Rotten Tomatoes review sentiment analysis, such as SVM+TF-IDF [8], NB+TF-IDF [9], and CNN [10]. On the other hand, other studies that use BERT apply BERT to things like hate speech detection and Bengali language sentiment analysis [6], [7]. Our research contribution is a BERT-based transformer model superior to SVM+TF-IDF, NB+TF-IDF, and CNN in Rotten Tomatoes review sentiment analysis.

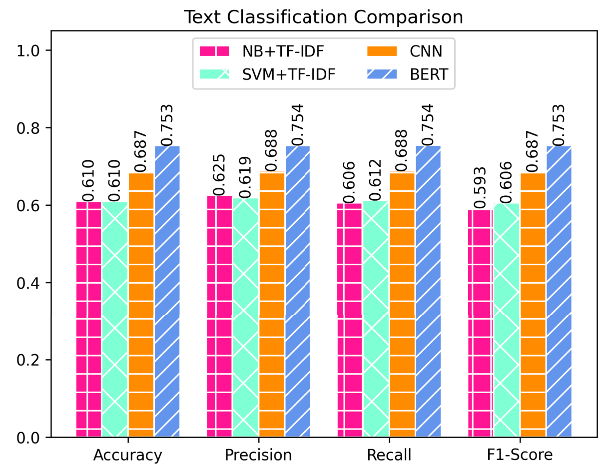

Fig. 5. The accuracy, precision, recall, and f1-score comparison of four sentiment analysis models.

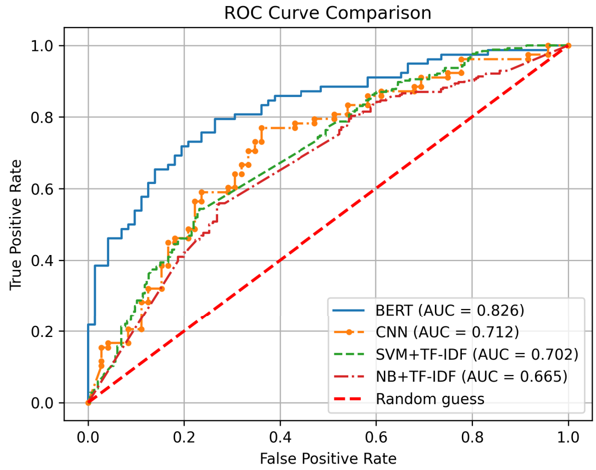

Fig. 6. The ROC and AUC comparison of four sentiment analysis models.

IV. CONCLUSION

Our research problem and summary for the research gap from previous research is to use BERT on sentiment analysis for Rotten Tomatoes reviews. We succeeded in implementing a model for Rotten Tomatoes Review Sentiment Analysis using a pre-trained transformer-based BERT. Our research is limited to using SVM+TF-IDF, NB+TF-IDF, and CNN as benchmarks. Our use of DistilBERT is proven to be more lightweight than the original BERT model. In the training optimization process, we learned that AdamW is more effective than Adam optimizer. Our test results show that the BERT model performance is better than the other three models with accuracy, precision, recall, and F1-score are 0.753, 0.754, 0.754, and 0.753, respectively. Finally, BERT is also the best in terms of AUC, with a value of 0.826. For future work, we propose to develop this discovery into an application that can give a positive or negative assessment of a film while adding text summarization.

ACKNOWLEDGEMENT

We thank Bo Pang and Lillian Lee for sharing Rotten Tomatoes Dataset in Huggingface. Our paper can become useful as the dataset is useful for us.

REFERENCES

[1] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, "Transformer in transformer," *Advances in Neural Information Processing Systems*, vol. 34, pp. 15908–15919, 2021.

[2] J. Qadir, "Engineering education in the era of chatgpt: Promise and pitfalls of generative ai for education," 2022.

[3] J. Jia, X. Chen, A. Yang, Q. He, P. Dai, and M. Liu, "Link of transformers in cv and nlp: A brief survey," in *2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI)*, pp. 735–743, IEEE, 2022.

[4] J. Bai, Y. Wang, Y. Chen, Y. Yang, J. Bai, J. Yu, and Y. Tong, "Syntax-bert: Improving pre-trained transformers with syntax trees," *arXiv preprint arXiv:2103.04350*, 2021.

[5] N. J. Prottasha, A. A. Sami, M. Kowsher, S. A. Murad, A. K. Bairagi, M. Masud, and M. Baz, "Transfer learning for sentiment analysis using bert based supervised fine-tuning," *Sensors*, vol. 22, no. 11, p. 4157, 2022.

[6] M. Mozafari, R. Farahbakhsh, and N. Crespi, "A bert-based transfer learning approach for hate speech detection in online social media," in *Complex Networks and Their Applications VIII: Volume 1 Proceedings of the Eighth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2019 8*, pp. 928–940, Springer, 2020.

[7] K. I. Islam, M. S. Islam, and M. R. Amin, "Sentiment analysis in bengali via transfer learning using multi-lingual bert," in *2020 23rd International Conference on Computer and Information Technology (ICCIT)*, pp. 1–5, IEEE, 2020.

[8] K. Soni, P. Yadav, *et al.*, "Comparative analysis of rotten tomatoes movie reviews using sentiment analysis," in *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 1494–1500, IEEE, 2022.

[9] P. Frangidis, K. Georgiou, and S. Papadopoulos, "Sentiment analysis on movie scripts and reviews: Utilizing sentiment scores in rating prediction," in *Artificial Intelligence Applications and Innovations: 16th IFIP WG 12.5 International Conference, AIAI 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings, Part I 16*, pp. 430–438, Springer, 2020.

[10] H. Sankar, V. Subramaniyaswamy, V. Vijayakumar, S. Arun Kumar, R. Logesh, and A. Umamakeswari, "Intelligent sentiment analysis approach using edge computing-based deep learning technique," *Software: Practice and Experience*, vol. 50, no. 5, pp. 645–657, 2020.

[11] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of the ACL*, 2005.

[12] S. F. Pane, R. Prastya, A. G. Putrada, N. Alamsyah, and M. N. Fauzan, "Reevaluating synthesizing sentiment analysis on covid-19 fake news detection using spark dataframe," in *2022 International Conference on Information Technology Systems and Innovation (ICITSI)*, pp. 269–274, IEEE, 2022.

[13] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[14] A. G. Putrada, I. D. Wijaya, and D. Oktaria, "Overcoming data imbalance problems in sexual harassment classification with smote," *International Journal on Information and Communication Technology (IJoICT)*, vol. 8, no. 1, pp. 20–29, 2022.

[15] M. L. Siddiq and J. C. Santos, "Bert-based github issue report classification," in *2022 IEEE/ACM 1st International Workshop on Natural Language-Based Software Engineering (NLBSE)*, pp. 33–36, IEEE, 2022.

[16] F. Roemer, M. Haardt, and G. Del Galdo, "Analytical performance assessment of multi-dimensional matrix-and tensor-based esprit-type algorithms," *IEEE Transactions on Signal Processing*, vol. 62, no. 10, pp. 2611–2625, 2014.

[17] I. Ofeidis, D. Kiedanski, and L. Tassiulas, "An overview of the data-loader landscape: Comparative performance analysis," *arXiv preprint arXiv:2209.13705*, 2022.

[18] M. Prajwal and M. A. Kumar, "Legal text analysis using pre-trained transformers," in *Advanced Machine Intelligence and Signal Processing*, pp. 493–504, Springer, 2022.

[19] R. Silva Barbon and A. T. Akabane, "Towards transfer learning techniques—bert, distilbert, bertimbau, and distilbertimbau for automatic text classification from different languages: A case study," *Sensors*, vol. 22, no. 21, p. 8184, 2022.

[20] S. Ghosh, S. Pratihar, S. Chatterji, and A. Basu, "Matching of hand-drawn flowchart, pseudocode, and english description using transfer learning," *Multimedia Tools and Applications*, pp. 1–29, 2023.

[21] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[22] V. Lodhi, A. Biswas, D. Chakravarty, and P. Mitra, "A study of deep learning approaches and loss functions for abundance fractions estimation," in *2021 11th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pp. 1–5, IEEE, 2021.

[23] Y. Wu, L. Liu, J. Bae, K.-H. Chow, A. Iyengar, C. Pu, W. Wei, L. Yu, and Q. Zhang, "Demystifying learning rate policies for high accuracy training of deep neural networks," in *2019 IEEE International conference on big data (Big Data)*, pp. 1971–1980, IEEE, 2019.

[24] A. I. Kadhim, "Term weighting for feature extraction on twitter: A comparison between bm25 and tf-idf," in *2019 international conference on advanced science and engineering (ICOASE)*, pp. 124–128, IEEE, 2019.

[25] M. Ameliasari, A. G. Putrada, and R. R. Pahlevi, "An evaluation of svm in hand gesture detection using imu-based smartwatches for smart lighting control," *JURNAL INFOTEL*, vol. 13, no. 2, pp. 47–53, 2021.

[26] B. A. Fadillah, A. G. Putrada, and M. Abdurohman, "A wearable device for enhancing basketball shooting correctness with mpu6050 sensors and support vector machine classification," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, 2022.

[27] A. G. Putrada, M. Abdurohman, D. Perdana, and H. H. Nuha, "Cima: A novel classification-integrated moving average model for smart lighting intelligent control based on human presence," *Complexity*, vol. 2022, 2022.

[28] A. G. Putrada, M. Abdurohman, D. Perdana, and H. H. Nuha, "Machine learning methods in smart lighting towards achieving user comfort: A survey," *IEEE Access*, 2022.

[29] A. Ali, W. Samara, D. Alhaddad, A. Ware, and O. A. Saraereh, "Human activity and motion pattern recognition within indoor environment using convolutional neural networks clustering and naive bayes classification algorithms," *Sensors*, vol. 22, no. 3, p. 1016, 2022.

[30] M. Ghobakhloo and M. Ghobakhloo, "Design of a personalized recommender system using sentiment analysis in social media (case study: banking system)," *Social Network Analysis and Mining*, vol. 12, no. 1, p. 84, 2022.

[31] M. R. S. Erwin, A. G. Putrada, and M. A. Triawan, "Deteksi hama ulat pada tanaman selada berbasis aquaponic menggunakan cnn (convolutional neural network)," *eProceedings of Engineering*, vol. 8, no. 5, 2021.

[32] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1d convolutional neural networks and applications: A survey," *Mechanical systems and signal processing*, vol. 151, p. 107398, 2021.

[33] Z. Tao, C. XiaoYu, L. HuiLing, Y. XinYu, L. YunCan, and Z. XiaoMin, "Pooling operations in deep learning: From "invariable" to "variable"," *BioMed Research International*, vol. 2022, 2022.

[34] A. Thakur and A. Konde, "Fundamentals of neural networks," *International Journal for Research in Applied Science and Engineering Technology*, vol. 9, pp. 407–26, 2021.

[35] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, *et al.*, "Recent advances in convolutional neural networks," *Pattern recognition*, vol. 77, pp. 354–377, 2018.

[36] A. G. Putrada and D. Perdana, "Improving thermal camera performance in fever detection during covid-19 protocol with random forest classification," in *2021 International Conference Advancement in Data Science, E-learning and Information Systems (ICADEIS)*, pp. 1–6, IEEE, 2021.

[37] A. G. Putrada and M. Abdurohman, "Anomaly detection on an iot-based vaccine storage refrigerator temperature monitoring system," in *2021 International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA)*, pp. 75–80, IEEE, 2021.

[38] A. G. Putrada, N. Alamsyah, S. F. Pane, and M. N. Fauzan, "Xgboost for ids on wsn cyber attacks with imbalanced data," in *2022 International Symposium on Electronics and Smart Devices (ISESD)*, pp. 1–7, IEEE, 2022.

[39] C. Y. Lee and Y.-P. P. Chen, "New insights into drug repurposing for covid-19 using deep learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 11, pp. 4770–4780, 2021.