UNDERGRADUATE RESEARCH IN NATURAL AND CLINICAL SCIENCE AND TECHNOLOGY (URNCST) JOURNAL
Read more URNCST Journal articles and submit your own today at: https://www.urncst.com

PRIMARY RESEARCH OPEN ACCESS

# The Implementation of Deep Learning Algorithm with Gaussian Blur Data Preprocessing in Circular RNA Classification and Detection

Evint Leovonzko, BSc Student [1], Callixta F. Cahyaningrum, BSc Student [1]*, Rachmania Ulwani, BSc Student [1]

[1] Department of Chemistry, University of British Columbia, Vancouver, British Columbia, Canada V6T 1Z2

*Corresponding Author: callixta@student.ubc.ca

**Abstract**

**Introduction:** Circular RNAs (circRNAs) are increasingly recognized as key regulators of gene expression due to their unique closed-loop structure and involvement in various cellular processes. This study investigates the utilization of machine learning algorithms in predicting circRNA-disease associations.

**Methods:** This study proposes a novel deep learning approach leveraging artificial neural networks (ANN) for circRNA classification. The methodology involves data collection from circRNA databases, k-mers counting for feature extraction, Gaussian blur implementation for data smoothing, and ANN-based model training.

**Results:** Evaluation of the trained models based on precision, recall, and f1-score metrics shows an overall accuracy of 0.7511, with an average precision score of 0.7982, recall of 0.7511, and f1-score of 0.7637.

**Discussion:** The results indicate that our ANN-based algorithm effectively detects and classifies circRNA datasets with considerable accuracy. Compared to the algorithm from past research, our algorithm is also shown to have less computational power.

**Conclusion:** Comparative analysis demonstrates improved performance compared to previous algorithms, suggesting its potential for widespread implementation due to reduced computational requirements and simpler implementation.

**Keywords:** circular RNA; artificial neural network; classification; deep learning; Gaussian blur

## Introduction

Circular RNAs (circRNAs) represent a category of single-stranded RNA molecules characterized by the absence of 5′ caps and 3′ polyadenylated tails, forming a covalently closed continuous loop structure [1]. Recent research indicates that circRNAs play a distinct role in regulating human gene expression [1]. They are produced through noncanonical RNA splicing by back splicing or head-to-tail splicing (see Figure A1 in Appendix), which deviates from typical splicing patterns [2]. This process can increase transcriptome diversity and contribute to the complexity of gene regulation and protein diversity [2]. The evolutionarily conserved sequence characteristics observed across diverse species, tissues, and developmental stages contribute to distinctive patterns of circRNAs [1]. These patterns play an important role in governing post-transcriptional gene regulation [1].

CircRNAs are emerging as valuable biomarkers due to their high stability and tissue-specific expression patterns, making them reliable biomarkers due to their consistent levels in different biological samples [3]. CircRNAs can be released from cells, including via exosomes, allowing for their detection in bodily fluids like blood or urine, enabling non-invasive diagnostic approaches [3]. Altered circRNA levels, such as circ_0004277 in acute myeloid leukemia, can indicate disease presence, progression, and response to treatment, highlighting their importance as biomarkers for various diseases [3]. These findings emphasize the various uses of circRNAs as biomarkers for disease.

Nowadays, machine learning has been utilized in bioinformatics fields to identify the regulatory interaction of cirRNAs [4], offering the potential for disease detection. Researchers examined databases and tools dedicated to circRNA-disease studies and comprehensively reviewed computational methods, including network propagating-based, path-based, matrix factorization-based, deep learning-based, and other machine learning approaches [5]. These computational methods significantly expedite the process of identifying potential circRNA-disease associations. However, the accuracy of these predictions is constrained due to the insufficient number of experimentally confirmed associations in the reference dataset [5].

The rapid development of machine learning has enhanced the prediction of circRNA-disease associations. For instance, Wang et al. (2020) introduced a machine-learning framework that incorporates Gradient Boosting

Decision Tree (GBDT) to identify circRNAs from RNA-sequencing data, with a high degree of accuracy [6]. Xiao et al. (2019) proposed an integrated computational framework, in which a weighted low-rank approximation optimization algorithm with dual-manifold regularizations was developed to identify disease-associated circRNAs [7].

Current models for predicting circRNA-disease associ ations are complex, despite the simplicity of circRNAs. This complexity, driven by intricate circRNA datasets, can lead to overemphasis on irrelevant details or noise. For instance, Yuan et al. (2021) proposed a simplified alternative model to identify circRNA-RBP (circular RNA binding protein) interactions called ResNet-18 [8]. However, this ResNet-18 model used a large model size, resulting in greater incidences of overfitting compared to its simplified model. This overfitting can result in a lack of generalization ability when the model encounters new data [9]. Conversely, simpler models are less susceptible to overfitting [10] and easier to interpret [11], resulting in more reliable performance, particularly in situations where the dataset is limited or contains noise.

While previous studies have used complex compu tational models to predict circRNA-disease associations, this study proposes a simpler model using Artificial Neural Networks (ANN) (see Figure A2 in Appendix). ANN is a simplified type of algorithm used in the Deep Learning approach which can provide high accuracy and reliability, as well as generalization to the new data [12]. Due to this, these simpler models might be faster to train and execute large datasets, making them more feasible for analysis. This study developed a suitable algorithm for designing and implementing a deep-learning model for circRNA classification with high accuracy. The model's performance was evaluated using precision, recall, and f1-score. Therefore, the accuracy of the proposed deep learning model for circRNA classification can be investigated.

## Methods

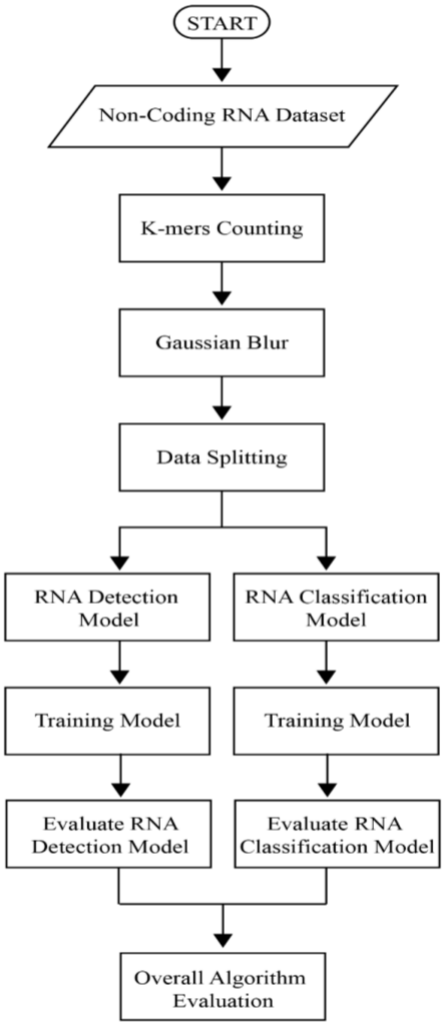


**Figure 1.** Overall Methodology Flowchart. Created with Adobe Illustrator.

This study aimed to create an algorithm based on artificial neural networks to detect and classify models. Overall, the method was divided into several parts: data collection, vectorization, preprocessing, model training, and algorithm evaluation. The algorithm was also divided into two parts, RNA detection, and RNA classification. All the available code may be found here: https://github.com/Evintkoo/circrna_classification.

Dataset Collection

Data was collected from circAtlas (https://ngdc.cncb.ac.cn/circatlas/links1.php) and processed with Python programming language. The data provided by circAtlas has already undergone the process of cleaning and is therefore readily available for use. The data consisted of the sequences of non-coding circular RNA & non-coding non-circular RNA (mRNA). It was converted into a vector before being processed with the k-mers counting.

Counting K-Mers

K-mer counting is the method of counting the number of subsequences with a length of "k" within a set of RNA sequences dataset, where "k" is a positive integer [13]. The length was chosen by the user (k = 4) based on this formula (where l = length of sequences) [14, 15]:

$$k = \log_k(average\ of\ l)$$

For k = 4, there are 4-mers obtained, e.g., {ACGT, GTAA, CGTT} which then can be processed with the following algorithm.

Gaussian Blur Implementation

Blurring is one of the techniques in data processing to smoothen the collected values in the dataset. This study implemented a Gaussian function to smoothen the vectorized non-coding RNA sequence. The data was reprocessed with the Gaussian blur to improve the accuracy [16], using this formula:

$$f(x) = 1/\left(\sigma sqrt(2\pi)\right)e^{-1/2((x-\mu)/\sigma)^2}$$

The value of sigma was 1 because it is often chosen as it provides a moderate amount of blurring, computational efficiency, and flexibility for adjustment.

Data Splitting

Then, the data was split using a ratio of 80:20, which means 80% of the data is for training and 20% for validation data [17]. This also means that the data for validation was not used to train the model to increase the reliability of the evaluation score.

Model Training

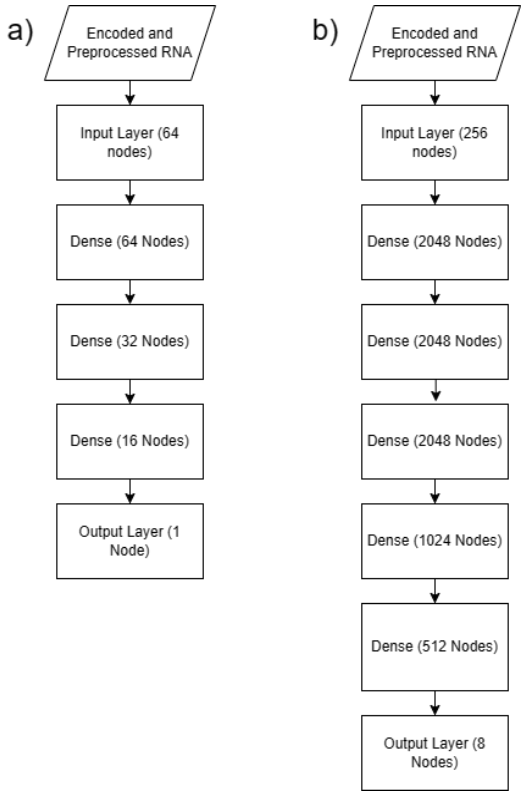


**Figure 2.** a) Detection Model Architecture and b) Classification Model Architecture. Created with https://www.drawio.com/.

The Artificial Neural Network (ANN) was proposed for the model training, in which the training process starts by assigning arbitrary initial connection weights, then updated until an acceptable training accuracy is achieved [18]. The proposed multi-classification neural network structure is arranged into an input layer (length: 256 nodes), 5 hidden layers, and an output layer. Every hidden layer was followed by a dropout layer to prevent overfitting [19] and a Leaky Rectified Linear Unit (ReLU) for adding non-linearity in the neurons [20]. However, there is no dropout layer after layer 5 as it did not provide additional benefits to the model's performance. The output layer is followed by the log softmax activation function that normalizes the output values into probabilities.

Artificial Neural Network was implemented for both detection and classification models. Dropout layer was added in each layer of the classification model Leaky ReLu activation function was added for each layer in the classification and detection model.

**Table 1.** Hyperparameter for Model Training

| Parameter | Detection Model | Classification Model |
|---|---|---|
| **Batch Size** | 64 | 64 |
| **Learning Rate** | $10^{-4}$ | $10^{-4}$ |
| **Optimizer Function** | Adam Optimizer | Adam Optimizer |
| **Loss Function** | Binary Cross Entropy | Negative Log Likelihood |
| **Accuracy Metric** | Binary Accuracy | Multiclass Accuracy |
| **Total Epoch** | 50 | 5000 |

Evaluation

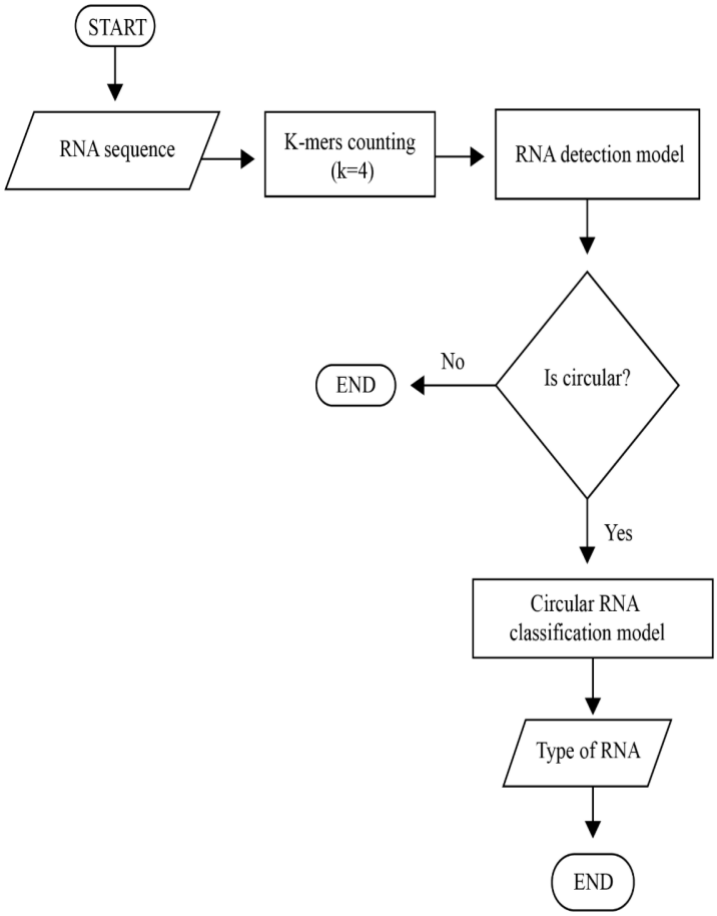


**Figure 3.** Flowchart Diagram for Combined Deep Learning Algorithm. Created with Adobe Illustrator.

Both models were combined as Figure 3 described, 20 percent of the data was used to test the model performance with precision (positive predictive value), recall, and f1-score for each class. Precision is calculated as the ratio of correctly predicted positive instances to the total predicted positives, focusing on the accuracy of positive predictions made by a model, *precision = True Positive/*(*True Positive + False Positive*).

Recall is determined as the ratio of correctly predicted positive instances to the total actual positives, emphasizing the model's ability to capture all relevant positive instances, *recall = True Positive/*(*True Positive + False Negative*).

The F1-score, a harmonic mean, provide a balance between precision and recall by considering both false positives and false negatives, $f1 - score = 2 \times precision \times recall / (precision + recall)$ [21].

These evaluations will provide a comprehensive understanding of the model's performance and its effectiveness in positive predictions.

**Results**

Figure 4 shows the training and validation loss function over the epoch of the detection model. Overall, the training and validation loss value shows a decreasing trend along with the increasing training model. The figure also showed that both training and validation loss values had a similar trend until the end of the training process. As shown, at the beginning there is a sudden drop of both loss values, while continued with a slow decrease of loss value.

Figure 5 shows the training and validation accuracy score of the detection model for each epoch number. The figure shows a rapid increase in training and validation accuracy scores at the beginning of the training process (from 0.7025; 0.87019 to 0.97992; 0.98098). The figure above also showed that both training and validation accuracy scores were increasing at roughly the same rate. The increasing trend slowly drops along with the number of epoch training.

Figure 6 shows the validation loss for each epoch of the training process for the classification model. As shown above, there was a decreasing trend for both training and validation loss along with the increasing number of epochs. However, on the 125th epoch, the decreasing trend of validation loss was changed into an increasing trend along with the increase of the number of epochs, while the training loss had a slower rate of decreasing value.

Figure 7 shows the training and validation accuracy of the training model for the classification model. The graph shows that the training and validation loss had a slow increment trend at the small epoch number, while at the 15th epoch, there is a trend change that shows a logarithmic increase up until the end of the training process. On top of that, both validation and training accuracy increased with similar rates and values up until the end of the training process.

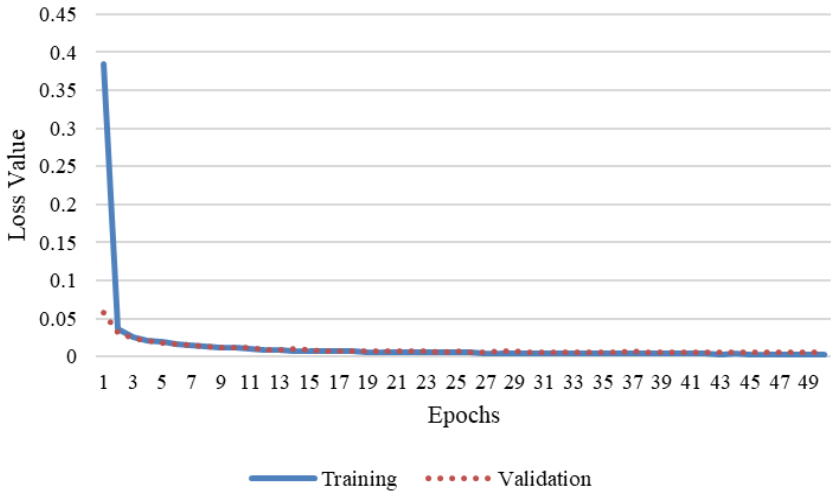


**Figure 4.** Train and Validation Loss per Epoch for the Detection Model. Created with Microsoft Excel.

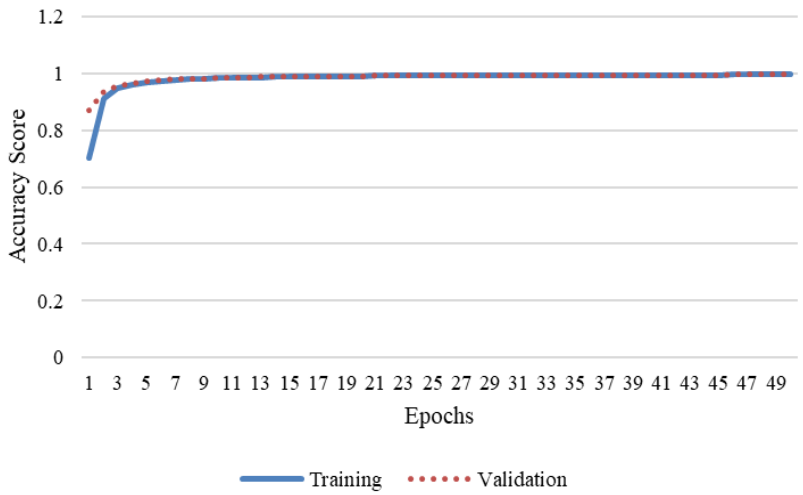


**Figure 5.** Train and Validation Accuracy per Epoch for Detection Model. Created with Microsoft Excel.

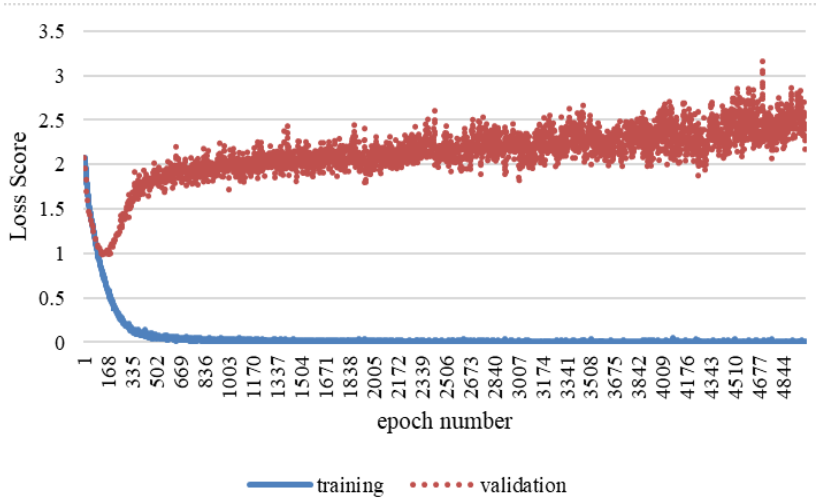


**Figure 6.** Train and Validation Loss per Epoch for Classification Model. Created with Microsoft Excel.

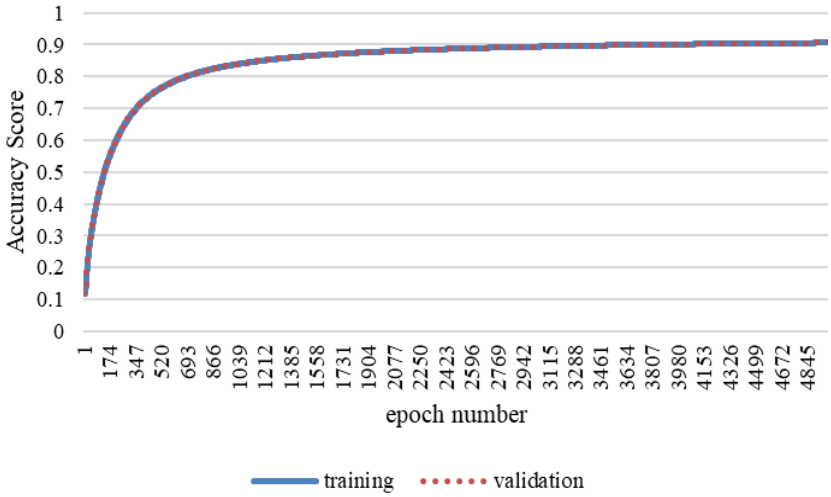


**Figure 7.** Train and Validation Accuracy per Epoch for Classification Model. Created with Microsoft Excel.

**Table 3.** Total Parameter on Prediction and Classification Model

| Model Type | Total Parameter | Training Accuracy | Validation Accuracy | Time per Prediction per RNA |
|---|---|---|---|---|
| **Detection Model** | 23233 | 0.9954 | 0.9954 | 0.047 ms |
| **Classification Model** | 11546120 | 0.9059 | 0.9059 | 0.093 ms |

The algorithm consisted of two parts: a detection model and a classification model. The detection model performed well, with a low loss score of 0.002 on training and 0.005 on validation, and high accuracy of 99.54% for both training and validation. The classification model showed decent progress, with a training loss of 0.001, validation loss of 2.36, and similar training and validation accuracy of around 90.59%.

**Table 4.** Accuracy, Precision, and F1-Score Table of Various Proposed Model

| Method | Accuracy | Precision | F1-Score |
|---|---|---|---|
| ***Our algorithm*** | 0.7511 | 0.7982 | 0.7637 |
| **SVM [22]** | 0.7328 | 0.7742 | 0.7921 |
| **RF [22]** | 0.7186 | 0.7393 | 0.7918 |
| **Att-CNN [22]** | 0.7264 | 0.7452 | 0.7696 |
| **Att-RNN [22]** | 0.7030 | 0.7189 | 0.7541 |
| **PredcircRNA [22]** | 0.5696 | 0.6218 | 0.5577 |
| **nRC [22]** | 0.7410 | 0.7662 | 0.8039 |
| **circDeep [22]** | 0.6140 | 0.7495 | 0.7229 |
| **DeepCirCode [22]** | 0.8129 | 0.9271 | 0.8365 |
| **JEDI [22]** | 0.8654 | 0.9074 | 0.8909 |

The overall algorithm showed that the accuracy of the proposed algorithm was 0.7511, with the average precision score of 0.7982, recall score of 0.7511, and f1-score of 0.7637. Compared to other similar methods such as CircDeep, the proposed method was evaluated with a higher score, while still relatively less accurate than DeepCirCode. When compared with other types of neural networks, such as Att-CNN and Att-RNN, the proposed method demonstrates slightly higher evaluation scores than both deep learning algorithms.

**Table 5.** Model Runtime

| Method | Time per Prediction per RNA |
|---|---|
| ***Our method*** | 0.14 ms |
| **SVM [22]** | 0.84 ms |
| **RF [22]** | 0.62 ms |
| **PredcircRNA [22]** | 1.29 ms |
| **JEDI [22]** | 4.87 ms |
| **DeepCircCode [22]** | 6.74 ms |
| **nRC [22]** | 7.22 ms |
| **Att-CNN [22]** | 23.67 ms |
| **Att-RNN [22]** | 91.35 ms |
| **circDeep [22]** | >1s |

The proposed method's runtime was significantly faster than other existing algorithms, taking only 0.1 milliseconds to predict one RNA. It was at least ten times faster than similar models like DeepcirCode and circDeep, six times faster than SVM, four and a half times faster than RF, and less than a hundredth of the time needed by different deep learning algorithms such as Att-CNN and Att-RNN.

**Discussion**

The objective of this experiment was to create a machine-learning model that can detect and classify different types of circular RNA with a high accuracy score by using an artificial neural network. The result of this experiment shows that the designed artificial neural network was able to detect and classify the circular RNA dataset with the value of evaluation score.

The similarity between both training and validation evaluation scores shows that there was not an underfitting and overfitting issue on the detection model. However, the high difference between the end result of training and validation loss in the classification model showed an indicator of overfitting that occurred in the training process. The similarity between the training and validation scores in terms of accuracy did not indicate an overfitting result, which contradicted the loss evaluation of the model. Table 3 also shows the number of artificial neural network parameters in both detection and classification models. The number of parameters of a model shows how complex and large the model is to be implemented. The first model shows that the number of parameters is relatively small. The first model would likely be a simple and small model that could be deployed with only CPU power since it only needs smaller amounts of computing power. On the other hand, the classification model would likely have a higher computational power to classify the circular RNA. Based on the computational power requirement, the proposed algorithm would have an advantage while detecting the RNA, since the algorithm would not use the classification model if the result of the detection model shows that it was not a circular RNA.

By comparing the evaluation score, the proposed method was also shown to have a better accuracy compared to SVM and RF, which showed that the proposed algorithm has a clear advantage in classifying circular RNA compared to different algorithms that have been tested. The results suggest that the proposed algorithm's architecture and training process contribute to its superior performance in identifying circular RNA sequences. Further research could explore the specific features of the algorithm that enable it to outperform other machine learning and deep learning methods in this task. Additionally, future studies could investigate the potential applications of this algorithm in various fields, such as biomedical research and drug discovery, where accurate identification of circular RNA plays a crucial role.

The proposed algorithm would likely have less computing power and simpler implementation compared to the past research, which would make its implementation less costly and more efficient. The significant reduction in runtime suggests that the proposed method has the potential to be more widely adopted due to its computational efficiency. This could lead to faster and more cost-effective RNA prediction in various applications, such as biomedical research and drug discovery. Furthermore, the simplicity of the proposed algorithm's implementation may encourage more researchers and practitioners to utilize this method, fostering collaboration and advancements in the field of RNA prediction. The reduced computational requirements and streamlined implementation of the proposed algorithm could also make it more accessible to researchers with limited computational resources, enabling them to contribute to the field and accelerate progress in RNA prediction and related areas of study.

This study suggests that the implemented algorithm that was based on an artificial neural network resulted in a decent value of accuracy and was evaluated to have better scores compared to the other past research [22]. By conducting runtime analysis, the proposed method is also shown to have a better efficiency in time and algorithm complexity, which leads to less computational power. Hence, the proposed algorithm might have a greater accuracy, while also having less computational power which made the model best to be commercialized and implemented widely.

However, this study has several limitations, such as the limited circular RNA dataset that only used human non-coding circular RNA to train the model which might reduce the accuracy of the model in its implementation. Another possible limitation was the training result, especially on the classification model, which shows that there is an indication of model overfitting even though the implemented model already has dropout layers. On top of that, another vectorizer algorithm to convert RNA into vectors might be implemented to increase the accuracy of the model.

## Conclusions

This study aimed to create an algorithm based on neural networks to detect and classify non-coding circular RNA with a high accuracy score. This study implemented an ANN as the detection and classification model and evaluated its accuracy, precision, recall, and f1-score. The results demonstrate that the proposed method was comparable to other similar algorithms. However, our algorithm was determined to have a better evaluation score compared to most other existing algorithms. In conclusion, the proposed algorithm was able to detect and classify the non-coding circular RNA with a high accuracy score.

## List of Abbreviations Used

RNA: ribonucleic acid
circRNAs: circular RNAs
NSCLC: non-small cell lung cancer
ncRNAs: non-coding RNAs
GBDT: gradient boosting decision tree
ANN: artificial neural networks
ReLU: leaky rectified linear unit
CPU: central processing unit
SVM: support vector machines
RF: random forest
CNN: convolutional neural networks
RNN: recurrent neural networks
Att-CNN: attentive-CNN
Att-RNN: attentive-RNN
nRC: non-coding RNA classifier
JEDI: junction encoder with deep interaction

## Conflicts of Interest

The authors declare that they have no conflict of interests.

**Ethics Approval and/or Participant Consent**
This study did not require ethics approval and/or participant consent.

**Authors' Contributions**
EL: made contributions to study design and planning, implemented the proposed methods and algorithm, collected and analyzed data, drafted the manuscript, and gave final approval of the version to be published.
CFC: made substantial contributions to the design of the study, drafted the manuscript, assisted analysis of data, revised the manuscript critically, and gave final approval of the version to be published.
RU: made substantial contributions to the design of the study, drafted the manuscript, revised the manuscript critically, and gave final approval of the version to be published.

**Acknowledgements**
The authors acknowledge the guidance and support of their mentor, Amel Sassi. The authors also acknowledge The URNCST Journal for providing the opportunity to complete the mentorship program.

**Funding**
This study was not funded.

**References**
[1] Greene J, Baird A-M, Brady L, Lim M, Gray SG, McDermott R, et al. Circular RNAs: Biogenesis, function and role in human diseases. Frontiers in Molecular Biosciences. 2017 Jun 6;4. https://doi.org/10.3389/fmolb.2017.00038
[2] Sibley CR, Blazquez L, Ule J. Lessons from non-canonical splicing. Nature Reviews Genetics. 2016 May 31;17(7):407–21. https://doi.org/10.1038/nrg.2016.46
[3] Allegra A, Cicero N, Tonacci A, Musolino C, Gangemi S. Circular RNA as a novel biomarker for diagnosis and prognosis and potential therapeutic targets in multiple myeloma. Cancers. 2022 March 27. 14(7): 1700. https://doi.org/10.3390/cancers14071700
[4] Zhang G, Deng Y, Liu Q, Ye B, Dai Z, Chen Y, et al. Identifying circular RNA and predicting its regulatory interactions by machine learning. Frontiers in Genetics. 2020 July 21. 11:655. https://doi.org/10.3389/fgene.2020.00655
[5] Xiao Q, Dai J, Luo J. A survey of circular RNAs in complex diseases: Databases, tools and computational methods. Briefings in Bioinformatics. 2021 October 21. 23(1):bbab444. https://doi.org/10.1093/bib/bbab444
[6] Wang Y, Zhang X, Wang T, Xing J, Wu Z, Li W, et al. A machine learning framework for accurately recognizing circular RNAs for clinical decision-supporting. BMC Medical Informatics and Decision Making. 2020 Jul 9;20(S3):137-49. https://doi.org/10.1186/s12911-020-1117-0
[7] Xiao Q, Luo J, Dai J. Computational prediction of human disease- associated circRNAs based on manifold regularization learning framework. IEEE Journal of Biomedical and Health Informatics. 2019 Nov;23(6):2661–9. https://doi.org/10.1109/JBHI.2019.2891779
[8] Yuan L, Yang Y. DeCban: Prediction of circRNA-RBP interaction sites by using double embeddings and cross-branch attention networks. Frontiers in Genetics. 2021 Jan 22;11. https://doi.org/10.3389/fgene.2020.632861
[9] Pham HN, Triantaphyllou E. The impact of overfitting and overgeneralization on the classification accuracy in data mining. Soft Computing for Knowledge Discovery and Data Mining. 2008;391–431. https://doi.org/10.1007/978-0-387-69935-6_16
[10] Piotrowski AP, Napiorkowski JJ. A comparison of methods to avoid overfitting in neural networks training in the case of catchment runoff modelling. Journal of Hydrology. 2013 Jan;476:97–111. https://doi.org/10.1016/j.jhydrol.2012.10.019
[11] Ying X. An overview of overfitting and its solutions. Journal of Physics: Conference Series. 2019 Feb;1168: 022022. https://doi.org/10.1088/1742-6596/1168/2/022022
[12] Wazirali R, Yaghoubi E, Abujazar MS, Ahmad R, Vakili AH. State-of-the-art review on energy and load forecasting in microgrids using artificial neural networks, machine learning, and deep learning techniques. Electric Power Systems Research. 2023 Dec;225:109792. https://doi.org/10.1016/j.epsr.2023.109792
[13] Manekar SC, Sathe SR. A benchmark study of k-mer counting methods for high-throughput sequencing. GigaScience. 2018 Oct 22;7(12). https://doi.org/10.1093/gigascience/giy125
[14] Melsted P, Pritchard JK. Efficient counting of k-mers in DNA sequences using a bloom filter. BMC Bioinformatics. 2011 Aug 10;12(1). https://doi.org/10.1186/1471-2105-12-333
[15] Reimann C, Filzmoser P, Hron K, Kynčlová P, Garrett RG. A new method for correlation analysis of compositional (environmental) data – A worked example. Science of The Total Environment. 2017 Dec;607–608:965–71. https://doi.org/10.1016/j.scitotenv.2017.06.063

[16] Song C, Xu W, Wang Z, Yu S, Zeng P, Ju Z. Analysis on the impact of data augmentation on target recognition for UAV-based transmission line inspection. Complexity. 2020 Sept 30;2020:1–11. https://doi.org/10.1155/2020/3107450
[17] Joseph VR. Optimal ratio for data splitting. Statistical Analysis and Data Mining: The ASA Data Science Journal. 2022 Apr 4;15(4):531–8. https://doi.org/10.1002/sam.11583
[18] Gupta MK, Agarwal K, Prakash N, Singh DB, Misra K. Prediction of miRNA in HIV-1 genome and its targets through artificial neural network: A bioinformatics approach. Network Modeling Analysis in Health Informatics and Bioinformatics. 2012 Sept 6;1(4):141–51. https://doi.org/10.1007/s13721-012-0017-3
[19] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research. 2014 Jan 1;15(1):1929–58. http://jmlr.org/papers/v15/srivastava14a.html
[20] Borawar L, Kaur R. ResNet: Solving vanishing gradient in deep networks. Proceedings of International Conference on Recent Trends in Computing. 2023 March 21;600: 235–47. https://doi.org/10.1007/978-981-19-8825-7_21
[21] Hicks SA, Strümke I, Thambawita V, Hammou M, Riegler MA, Halvorsen P, et al. On evaluation metrics for medical applications of artificial intelligence. Scientific Reports. 2022 Apr 8;12(1):5979–88. https://doi.org/10.1038/s41598-022-09954-8
[22] Jiang J-Y, Ju CJ, Hao J, Chen M, Wang W. Jedi: Circular RNA prediction based on junction encoders and deep interaction among splice sites. Bioinformatics. 2021 Jul 1;37(Supplement_1):i289–98. https://doi.org/10.1093/bioinformatics/btab288

**Article Information**
Managing Editor: Jeremy Y. Ng
Peer Reviewers: Amel Sassi, Clara Rose Schott
Article Dates: Received Mar 31 24; Accepted May 31 24; Published Jul 04 24

**Citation**
Please cite this article as follows:
Leovonzko E, Cahyaningrum CF, Ulwani R. The implementation of deep learning algorithm with Gaussian blur data preprocessing in circular RNA classification and detection. URNCST Journal. 2024 Jul 04: 8(6).
https://urncst.com/index.php/urncst/article/view/601
DOI Link: https://doi.org/10.26685/urncst.601