International Student Research Review
Volume 2 Issue 1 May 2025
DOI: https://doi.org/10.53378/isrr.166

Enhancement of convolutional neural networks algorithm for application form using GlobalMaxPooling in document verification system

[1]Fabien Georgette M. Lapuz, [2]Michael D. Carlos, [3]Khatalyn E. Mata & [4]Mark Anthony S. Mercado

Abstract

This study focuses on improving Convolutional Neural Networks (CNNs) to automate document verification system by utilizing CNN's memory and computational complexity with adapting its structures to the specific characteristics of image data. Moreover, pooling is a crucial process for reducing the dimensionality of extracted features, a known key component of CNN Architecture. Thus, choosing the most appropriate pooling method is crucial across numerous computer vision architectures. Conventional pooling techniques like max pooling and average pooling have been extensively utilized for dimensionality reduction. However, both technique presents its own set of limitations such as loss of important details that are essential for tasks demanding high precision, while also diminishing the significance of key features by distributing attention across all values. This study presents the alternative pooling method of GlobalMaxPooling, aimed at capturing the most significant patterns and emphasizing critical patterns pertinent to document verification tasks. Using a dataset of 750 application forms, our results demonstrated an increase of significant improvement in detection accuracy, with the enhanced model achieving an accuracy of 77.2% compared to the existing model's initial 20% accuracy. Furthermore, these findings emphasize the importance of effective pooling methods thereby strengthening the model's capability for document verification, paving the way broader applications in automated systems requiring high precision and scalability.

Keywords: *automation, document verification, convolutional neural networks (CNN), pooling method*

Article History:

***Received**: March 16, 2025* | ***Revised**: May 24, 2025*
***Accepted**: May 28, 2025* | ***Published online**: May 31, 2025*

Suggested Citation:
Lapuz, F.G.M., Carlos, M.D., Mata, K.E. & Mercado, M.A.S (2025). Enhancement of convolutional neural networks algorithm for application form using GlobalMaxPooling in document verification system. *International Student Research Review*, 2(1), 108-123. https://doi.org/10.53378/isrr.166

About the authors:
[1]Corresponding author. Pamantasan ng Lungsod ng Maynila. College of Information Systems and Technology. 4th Year BSCS Student. fabiengeorgettelapuz@gmail.com
[2]Pamantasan ng Lungsod ng Maynila. College of Information Systems and Technology. 4th Year BSCS Student.
[3]Pamantasan ng Lungsod ng Maynila. College of Information Systems and Technology. Acting Dean.
[4]Pamantasan ng Lungsod ng Maynila. College of Information Systems and Technology. Faculty.

1. Introduction

Document verification plays an important role in ensuring safe transactions and preserving both legal compliance and critical information. Different fields such as finance, law, and identity verification rely on documents, both printed and electronic. There are different dangers that arise upon document verification (Shende et al., 2024: Rajapakshe et al., 2020); authentication of documents has been a significant challenge faced by different organizations, both public and private. Manual authentication of documents ultimately is prone to errors due to its reliance on human dependence. Errors include data entry typos, inappropriate or incomplete processing of data (Ogale, 2023).

Convolutional Neural Networks (CNN) is a class of deep neural networks that applies to analyze visual imagery. CNN can learn features from the input without removing the task. Some of the applications of CNN are image recognition, image classification, and speech recognition (Upretti, 2022; Khanday & Dadvandipour, 2020). Additionally, in convolution-based systems, pooling is an integral step in lowering the dimensionality of derived features. Reducing the number of variables in the value sequence reduces the dimension of the feature map. It converts basic feature descriptions into actionable information by keeping the information relevant and eliminating superfluous features. Pool operations provide several services. By removing the incomplete connections between the convolution layers, the pooling process gives some spatially modified representation while lowering the computational cost of the highest layers (Zafar et al., 2022).

Overfitting occurs when the gap between training and test error is too big. That is, the model is more complex than the actual problem, and it performs well on the training set but poorly on the test set (Xiao et al., 2021; Ying, 2019). In CNN, the pooling layer executes the down-sampling on the feature maps coming from the previous layer and produces new feature maps with a condensed resolution. Having an ideal pooling method is expected to extract only useful information and discard irrelevant details (Gholamalinezhad & Khosravi, 2020). Therefore, choosing the appropriate pooling method for image recognition plays a crucial role in reducing the spatial dimensions of feature maps and enhancing feature extraction more robust small variations or distortions in input data.

Applying an algorithm-based solution utilizing Global Max Pooling technique can enhance the current CNN algorithm and can serve as a sufficient aid in verifying documents for PLMAT applicants. This research proposed the enhancement of CNN can eliminate the problem overfitting in the CNN algorithm. This study argues that overfitted CNN can become too sensitive to noise and easily generalize unseen data. Similarly, CNN faces challenges in handling variations in document layout and complex features that results in suboptimal performance in document verification. Hence, improper implementation of pooling methods leads to a loss of highly representative values.

This study has several implications for the field of image validation and detection. It expedites user authentication process by providing a streamlined platform, reducing human intervention in the process. It also automatically detects inconsistencies, minimizes the risk of human error in verification, easily facilitates real-time analysis and provides immediate feedback on the submitted application forms. This study's findings will contribute to improving accuracy, efficiency, and fairness of document verification systems in academic contexts, resulting in a more beneficial environment for learning and academic performance.

2. Literature review

2.1. Document Verification

Document verification plays a pivotal role in preserving the integrity of critical and personal information. With the advancement of digital technology, personal documents such as birth certificates, and certificates of grades can easily be tampered without being detected (Shende et al., 2024; Aldwairi et al., 2023; Koshiry et al., 2023). For admissions, a lot of institutions and organizations require notification and verification of qualification as a prerequisite with the main objective to recognize the authenticity of a copy or digital document issued by another institution and detect forgeries (Aldwari et al., 2023). Important documents such as proof of education, and certificate of grades that are issued by educational institutions are examples of requirements that admission tests often require (Kumutha et al., 2022). Applicants will need to upload their documents within the designated time frame, and afterwards they will be evaluated wherein they will either be verified, or they need to comply with additional requirements. With over 51,000 applicants that applied for the last PLMAT 2023-2024, the verification is a time-consuming process. It is required to seek vast data for proper verification of documents.

Through the advancement of technology, there have been a few innovations to data verification such as the data verification and validation using blockchain in a clinical trial for breast cancer. A blockchain-based data management system was developed for clinical trials.

The researcher supplied random allocation produced by a computer. The contents of the app that participants used during the clinical trial were automatically assigned to the habit-B program or the control using the generated account (Hirano et al., 2020).

2.2. CNN for Data Verification

Compared to a fully connected neural network, CNN is more efficient in terms of memory and complexity (Upretti, 2022; Zhao et al., 2024; Zhen & Bărbulescu, 2024). It plays a pivotal role in adapting structures to image structures while extracting and classifying features. In deep learning, CNN has emerged as the most representative neural network, which primarily employed in image classification, segmentation, object detection, video processing, natural language processing, and speech recognition. It has been used to tackle complex visual tasks with high computation (Purwono et al., 2023).

Compared to other algorithms, CNN automatically identifies the relevant features without any human supervision. It has equivalent representations, sparse interactions, and parameter sharing (Alzubaidi et al., 2021). It is the machine learning algorithm that is often used in image recognition, especially the applications that deal with image data. With the success of recognizing digital images on a dataset called MNIST, CNN has continued to be applied on several applications (Rizky et al., 2023). When compared to RNN (Recurrent Neural Network), CNN is much more powerful since RNN includes less feature compatibility, specifically with RNN’s sensitivity to the exploding gradient and vanishing problems represent one of the main issues (Alzubaidi et al., 2021).

CNN is also one of the better options for implementation than Artificial Neural Network (ANN), due to its lesser parameters since CNN can be trained smoothly (Ahmed & Karim, 2020). CNN is widely being used in various domains due to its remarkable performance for image classification (Anton et al., 2021). When CNN compared to Long Short-Term Memory (LSTM), and Gated Current Links (GRUs) models for automatic document classification, CNN achieved the highest accuracy with 0.90 compared to the 0.77 accuracy of LSTM and 0.82 of GRU (Sun et al., 2019).

2.3. CNN’s Pooling Layer

CNN’s pooling layer plays a crucial role in reducing spatial dimensions and improving computational efficiency. Standard pooling operations such as max pooling or average pooling are not suitable for all data types. A study conducted by Zhao and Zhang (2024) shows the development of a custom pooling layer that can adaptively learn and extract relevant features from specific data sets entitled T-Max-Avg pooling layers. According to extensive experimental research using different datasets, the T-Max-Avg method has shown higher accuracy compared to Avg-TopK, maximum, and average pooling methods. It indicates that the T-Max-Avg method can more accurately capture feature information and provide better results during the model training process.

Furthermore, using the average pooling method, variables that are minimally representative and highly representative are handled equally. It makes sure that when there are values near zero, the outcome is also near zero and the dominant values do not get the values they deserve. When the highest value in a pool of data is a noisy pixel point, the selection of the highest representative value takes precedence over all other values, which can have a substantial effect on classification performance.

2.4. Problems Faced by CNN Algorithm

The CNN algorithm often faces the challenge of overfitting, this is the situation where the CNN model learns the statistical regularities specific to the training set and ends up memorizing the irrelevant noise, instead of learning the pattern, causing it to perform less on other datasets. Having an overfitted model is not generalizable to never-seen-before data (Yamashita et al., 2018). CNNs and other machine learning techniques are known to be vulnerable to overfitting. When training on training data, an overfitted learner does well, but when using new data, it does not perform as well as predicted. Overfitting is not just harmful because it causes performance on unseen data to drop unexpectedly (Xia, 2024).

Halim et al. (2023) conducted a study on hyperparameter tuning in CNN on image classification models regarding a case study of plant disease detection utilizing Google Colaboratory platform and AiSara tuning algorithm. The evaluation was measured by comparing the accuracy loss and accuracy metrics, noting that the performance of the CNN algorithm is greatly influenced by hyper-parameter selection. Having any small changes in the hyper-parameter values will affect the general CNN performance. Hence, careful parameter selection is a must during optimization scheme development (Alzubaidi et al., 2021).

CNNs serve to autonomously acquire a hierarchy of features for classification, rather than relying on manual feature engineering. A hierarchy of feature maps is established by sequentially convolving the input image with learned filters to do this. The hierarchical approach enables upper layers to acquire more complex properties that are invariant to distortion and translation (Taye, 2023). Moreover, with the amount of depth of CNN, training has become challenging (Brillantes et al., 2019). To further enhance the effectiveness of CNN, a balanced approach to filter size selection, and regulation is important.

2.5. Theoretical Framework

Figure 1

Theoretical framework

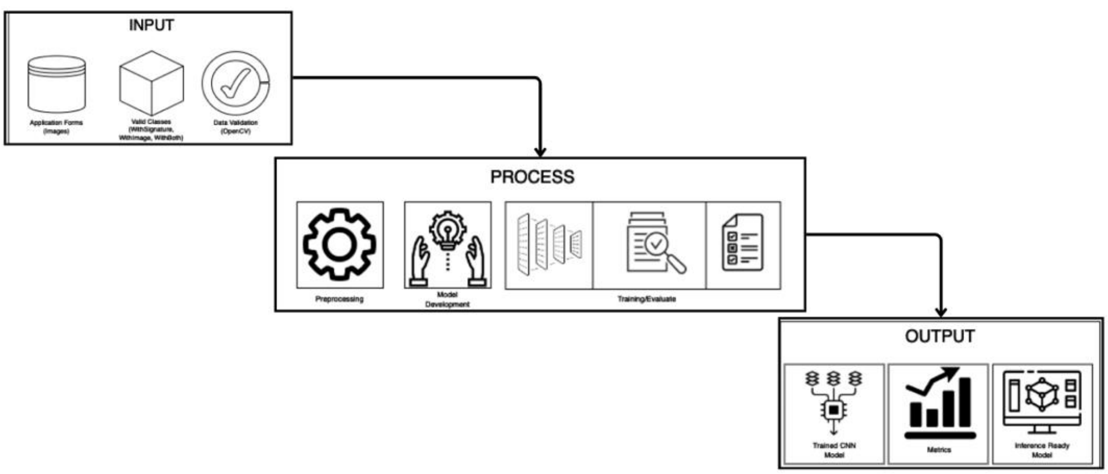

Figure 1 outlines the theoretical framework of this study composed of three main stages: input, process, and output. The input stage involves collecting application form images, validating data using predefined classes, and ensuring integrity through data validation techniques like OCR or Deep Learning-based methods. In the process stage, the data undergoes preprocessing to clean and transform it into a suitable format, followed by model development using deep learning, where training and evaluation refine the model for optimal performance. Finally, the output stage produces a trained CNN model, generates performance metrics, and prepares an inference-ready model for real-time predictions or deployment. This framework effectively captures the structured workflow of an image-based document verification system.

3. Methodology

3.1. Research Design

This study aims to develop and enhance the most recent CNN algorithm for image verification system. Furthermore, the research follows a structured pipeline consisting of dataset preparation, model development, training, evaluation, and model deployment.

The study utilized a systematic approach in creating application form datasets, ensuring a categorical approach to test the accuracy of the prediction of the enhanced image recognition algorithm. The research utilized a blank template of the application form and created a total of 750 datasets containing dummy data. 500 of the images were utilized in the training, and 250 were utilized for testing. The study utilized Microsoft Word’s draw feature in drawing each unique signature of the applicant forms. Meanwhile, Pinterest and Canva were used as main source for gathering formal photos of the applicant forms.

Figure 2

Sample datasets

Figure 2 illustrates the utilized datasets, arranged from left to right as follows: with_both, with_image_only, with_signature_only, and with_wrong_image. Simultaneously, a blank application form is employed for with_no_data to assist the model in identifying the characteristics of the blank application form. Overall, the datasets were meticulously assembled to represent a wide array of possible real-world situations, guaranteeing that the improved picture recognition algorithm is evaluated under diverse conditions. The study aims to evaluate the model's capacity to accurately categorize incomplete or erroneous application forms by introducing categories.

3.2 Model Development

A baseline CNN model with TensorFlow and Keras are used to create a baseline CNN model. The architecture is made up of five layers with ReLU activation functions, max-pooling layers, and a fully connected output layer with a softmax activation function for multiclass classification. Adam was used as the optimizer, with a learning rate of 0.0005, and the loss function was categorical cross-entropy, which is appropriate for multi-class classification problems. The model's architecture tries to strike a balance between complexity and computing efficiency, resulting in accurate but feasible training and deployment.

3.3 Training

The model is trained over 30 epochs with a batch size of 32. The dataset is fed into the model in mini batches, allowing the network to repeatedly change its parameters using backpropagation and the Adam optimizer. The training step guarantees that the model learns significant patterns from the dataset, resulting in lower classification mistakes over time. Early stopping and validation loss monitoring are used to avoid overfitting and achieve optimal generalization.

3.4 Evaluation

In the testing phase, the performance of an existing CNN model and an improved CNN model is assessed through their accuracy as determined by a confusion matrix. The matrix evaluates the accuracy, precision, recall, and F1 Score of the algorithm's proper recognition within the dataset.

Accuracy is used to measure the performance of the model. It is the ratio of Total correct instances to the total instances.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \; x \; 100$$

Precision is a measure of how accurate a model’s positive predictions are.

$$Precision = \frac{TP}{TP + FP} \; x \; 100$$

Recall measures the effectiveness of a classification model in identifying all relevant instances from a dataset.

$$Recall = \frac{TP}{TP + FN} x\ 100$$

F1-Score is used to evaluate the overall performance of a classification model. It is the harmonic mean of precision and recall,

$$F1\ Score = \frac{2\ x\ Precision\ x\ Recall}{Precision + Recall} x\ 100$$

4. Findings and Discussion

Figure 3

Performance evaluation of existing algorithm

Figure 3 shows the existing algorithm's performance in terms of accuracy (left) and loss (right) across numerous epochs of training, validation, and testing data. The left graph depicts training, validation, and test accuracy. The training and validation accuracies vary dramatically, indicating instability or overfitting. The test accuracy remains constant at approximately 0.5, indicating low generalization. The right graph displays the training, validation, and test loss, with the training loss lowering sharply at the start and remaining close to zero, while the validation and test losses remain relatively flat. This pattern shows that the model memorized the training data but did not improve generalization, which could be due to excessive model complexity, low data quality, or insufficient regularization.

Table 1

Performance metrics of existing algorithm

Category	Accuracy	Recall	Precision	F1-Score
Existing Algorithm	20%	20%	4%	7%

Table 1 indicates the performance metrics of the existing algorithm and how it varies across the accuracy, recall, precision, and F-1 score categories. Among these metrics, the algorithm achieved an accuracy of 20%. Meanwhile, the result of 20% recall score shows that the existing algorithm missed a large portion of the relevant cases. The 4% precision result indicates that the model generated a lot of false positives during prediction. Lastly, the F1-Score of 7% indicates the poor balance between precision and recall wherein the model failed to classify.

Figure 4

Performance evaluation of enhanced algorithm

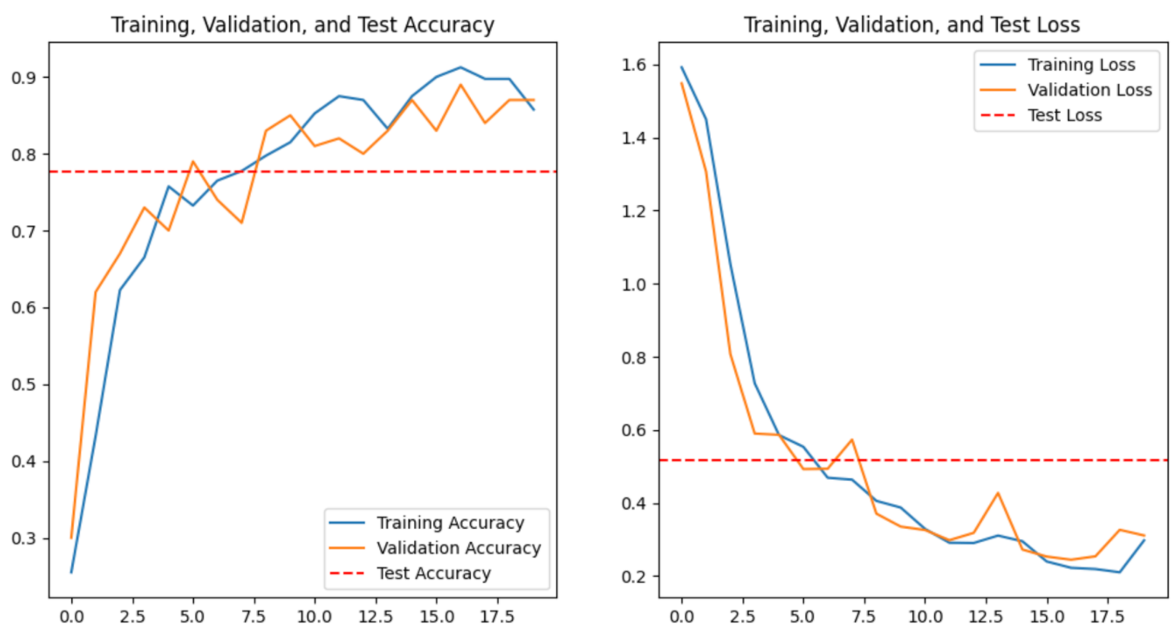

Figure 4 shows the enhanced algorithm's performance in terms of accuracy (left) and loss (right) across numerous epochs of training, validation, and testing data. The left graph demonstrates that training and validation accuracy improve steadily over time, achieving values more than 0.9 and 0.85, respectively, suggesting strong model learning. The test accuracy (red dashed line) remains consistent at 0.8, indicating good generalization. The right graph shows the training, validation, and test loss, with both training and validation losses continually decreasing, indicating effective learning. Although there are modest fluctuations in validation loss, the overall pattern is consistent with training loss. This indicates that the model is learning effectively with minimal overfitting, although additional testing on unseen data is required.

Table 2

Performance metrics of enhanced algorithm

Category	Accuracy	Recall	Precision	F1-Score
Enhanced Algorithm	77.2%	77.2%	67.8%	70.7%

Table 2 indicates the performance metrics of the enhanced algorithm and how it varies across the accuracy, recall, precision, and F-1 score categories. Among these metrics, the algorithm achieved an accuracy of 77.2%. Meanwhile, the result of 77% recall score shows that the existing algorithm missed a large portion of the relevant cases. The 67.8% precision result indicates that the model generated a lot of false positives during prediction. Lastly, the F1-Score of 70.7% indicates an increase between precision and recall wherein the model was able to classify the dataset accordingly.

Table 3

Comparative Performance Metrics of Existing and Enhanced Algorithm

Category	Accuracy	Recall	Precision	F1-Score
Existing Algorithm	20%	20%	4%	7%
Enhanced Algorithm	77.2%	77.2%	67.8%	70.7%
Increase in performance	57.2%	57.2%	63.8%	63.7%

Table 3 compares the performance metrics of the existing CNN algorithm with the enhanced CNN algorithm across accuracy, recall, precision, and F1-Score metrics. The existing algorithm attained an accuracy of 20%, whereas the enhanced algorithm advanced to 77.2%, representing a 57.2% enhancement in performance. This indicates a notable enhancement in the model's capacity to accurately categorize cases. The existing algorithm exhibited a recall of 20%, signifying it identified merely a limited fraction of the pertinent cases. The enhanced algorithm achieved a 77.2% recall, indicating a 57.2% improvement and enhanced capacity to identify pertinent instances.

As for precision, the existing algorithm attained a precision of 4%, while the enhanced algorithm achieved a precision of 67.8%, reflecting a 67.8% increase, indicating superior capability in predicting real positives and minimizing false positives. Ultimately, the existing algorithm exhibited an F1-Score of 7%, indicating a deficient equilibrium between precision and recall. The enhanced algorithm achieved a 70.7% F1-Score, reflecting a 63.7% improvement, indicating superior equilibrium between precision and recall, resulting in enhanced overall performance.

Figure 5

Comparative performance evaluation of enhanced algorithm

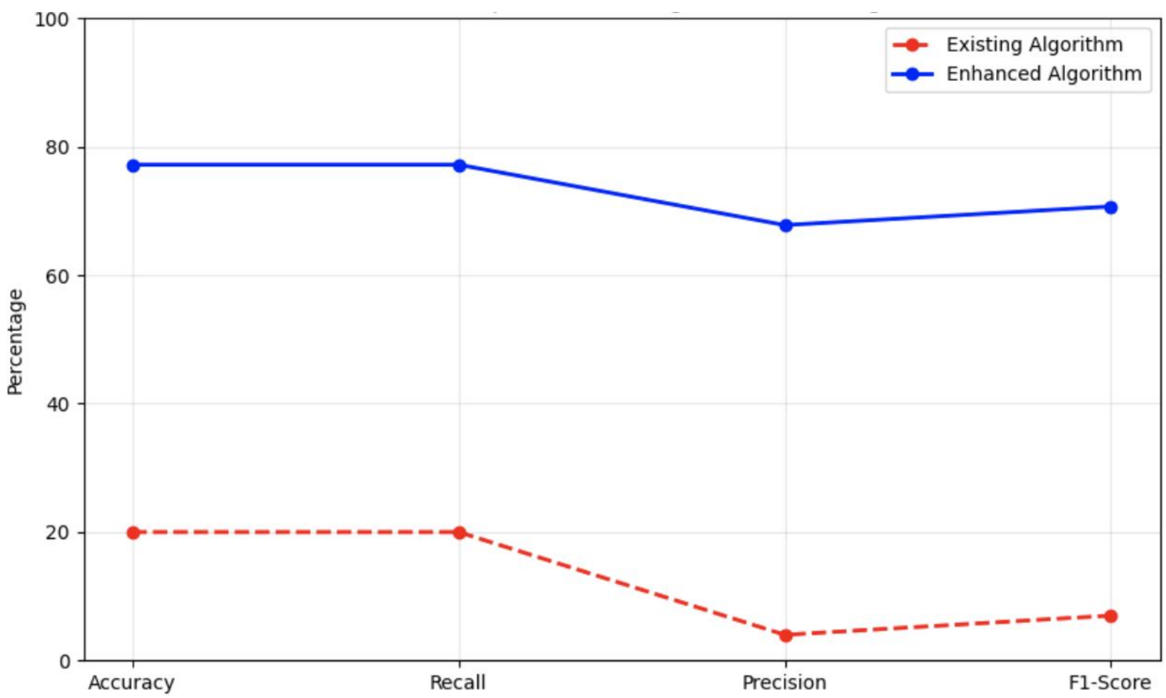

Figure 5 compares the performance of the existing algorithm (red, dashed line) and an enhanced algorithm (blue, solid line) based on four major assessment metrics: accuracy, recall, precision, and F1 score. The improved algorithm surpasses the previous system in all metrics, with accuracy and recall, while precision falls somewhat but stays relatively high. In comparison, the existing algorithm performs badly, maintaining approximately 20% accuracy and recall but dropping considerably lower in precision. The F1-score follows a similar pattern, highlighting the enhanced algorithm's better performance. This shows that the improved algorithm is far more effective and dependable than the existing.

5. Conclusion

The findings from the comparative study showed that with GlobalMaxPooling, the validation accuracy ranged increased from 20% to 77.2 %. This improvement was able to diminish the model's ability to generalize unseen data. Furthermore, the enhanced model was able to handle complex structures by focusing on the most important regions of the document, enhancing its classification accuracy and GlobalMaxPooling was able to reduce spatial dimensions of feature maps and preserve critical information in application forms.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was not supported by any funding.

AI Declaration

The authors declare the use of Artificial Intelligence (AI) in writing this paper. In particular, the authors used Quilbot and Chatgpt for correcting the grammar and the analysis of literatures. The authors take full responsibility in ensuring that research idea, analysis and interpretations are original work.

ORCID

Fabien Georgette Lapuz – https://orcid.org/0009-0002-4633-1414

References

Ahmed, W., & Karim, A. (2020). The impact of filter size and number of filters on classification accuracy in CNN. *2020 International Conference on Computer Science and Software Engineering (CSASE)*. https://doi.org/10.1109/csase48920.2020.9142089

Aldwairi, M., Badra, M., & Borghol, R. (2023). DocCert: Nostrification, document verification and authenticity blockchain solution. *2023 Fifth International Conference on Blockchain Computing and Applications (BCCA)*. https://doi.org/10.1109/bcca58897.2023.10338908

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of Deep Learning: Concepts, CNN Architectures, challenges, applications, Future Directions. *Journal of Big Data*, *8*(1). https://doi.org/10.1186/s40537-021-00444-8

Dishar, H. K., & Muhammed, L. A. (2023). A review of the overfitting problem in convolution neural network and remedy approaches. *Journal of Al-Qadisiyah for Computer Science and Mathematics*, *15*(2). https://doi.org/10.29304/jqcm.2023.15.2.1240

Gholamalinejad, H., & Khosravi, H. (2020, December). *(PDF) pooling methods in Deep Neural Networks, a review*. arXiv. https://doi.org/10.48550/arXiv.2009.07485

Gurucharan, M. (2022). *Basic CNN architecture: Explaining 5 layers of Convolutional Neural Network*. upGrad blog. https://www.upgrad.com/blog/basic-cnn-architecture/

Halim, A., Chow, C., Amabel, M., Achmad, S., & Sutoyo, R. (2023). The impact of hyperparameter tuning in convolutional neural network on Image Classification Model: A Case Study of Plant Disease Detection. *2023 5th International Conference on Cybernetics and Intelligent System (ICORIS)*. https://doi.org/10.1109/icoris60118.2023.10352209

Hirano, T., Motohashi, T., Okumura, K., Takajo, K., Kuroki, T., Ichikawa, D., Matsuoka, Y., Ochi, E., & Ueno, T. (2020). Data validation and verification using blockchain in a clinical trial for breast cancer: Regulatory sandbox. *Journal of Medical Internet Research*, *22*(6). https://doi.org/10.2196/18938

Koshiry, A.E., Eliwa, E., El-Hafeez, T. & Shams, M.Y. (2023). Unlocking the power of blockchain in education: An overview of innovations and outcomes. *Blockchain: Research and Applications,* 4(4), 100165. https://doi.org/10.1016/j.bcra.2023.100165

Khanday, O. M., & Dadvandipour, S. (2020). Convolutional neural networks and impact of filter sizes on image classification. *Multidiszciplináris Tudományok*, *10*(1), 55–60. https://doi.org/10.35925/j.multi.2020.1.7

Kumutha, K., Jayalakshmi, S., & SriPriya, P. (2022). Data security mechanism of hyperledger fabric and its application in academic documents verification system. *International Journal of Health Sciences*, 4394–4403. https://doi.org/10.53730/ijhs.v6ns3.6821

Pavlyshenko, B., & Stasiuk, M. (2024). Data augmentation in text classification with multiple categories. *Electronics and Information Technologies*, 25(25). https://doi.org/10.30970/eli.25.6

Purwono, P., Ma'arif, A., Rahmaniar, W., Fathurrahman, H. I., Frisky, A. Z., & Haq, Q. M. (2023). Understanding of convolutional neural network (CNN): A Review. *International Journal of Robotics and Control Systems*, *2*(4), 739–748. https://doi.org/10.31763/ijrcs.v2i4.888

Rajapakshe, M., Adnan, M., Dissanayaka, A., Guneratne, D. & Abeywardena, K. (2020). Multi-Format document verification system. *American Scientific Research Journal for Engineering, Technology, and Sciences*, 74(02), 48-60.

Rizky, A. F., Yudistira, N., & Santoso, E. (2023). Text recognition on images using pretrained CNN. *arXiv*. https://doi.org/10.48550/arXiv.2302.05105

Shende, A., Mullapudi, M. & Challa, N. (2024). Enhancing document verification systems: A review of techniques, challenges, and practical implementations. *International Journal of Computer Engineering and Technology*, 15(1), 16-25. https://doi.org/10.17605/OSF.IO/HVQ8E

Sun, X., Li, Y., Kang, H., & Shen, Y. (2019). Automatic document classification using Convolutional Neural Network. *Journal of Physics: Conference Series*, 1176, 032029. https://doi.org/10.1088/1742-6596/1176/3/032029

Upreti, A. (2022). Convolutional Neural Network (CNN). A Comprehensive Overview. https://doi.org/10.20944/preprints202208.0313.v3

Xia, Z. (2024). Overfitting of CNN model in CIFAR-10: Problem and solutions. *Applied and Computational Engineering*, *37*(1), 212–221. https://doi.org/10.54254/2755-2721/37/20230511

Xiao, M., Wu, Y., Zuo, G., Fan, S., Yu, H., Shaikh, Z. A., & Wen, Z. (2021). Addressing overfitting problem in deep Learning-Based solutions for next generation Data-Driven networks. Wireless Communications and Mobile Computing, 2021(1). https://doi.org/10.1155/2021/8493795

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional Neural Networks: An overview and application in radiology. *Insights Imaging*, 9, 611–629. https://doi.org/10.1007/s13244-018-0639-9

Ying, X. (2019). An overview of overfitting and its solutions. *Journal of Physics: Conference Series*, 1168, 022022. https://doi.org/10.1088/1742-6596/1168/2/022022

Zhao, L., & Zhang, Z. (2024). An improved pooling method for Convolutional Neural Networks. *Scientific Reports*, *14*(1).

Zhao, X., Wang, L., & Zhang, Y. (2024). A review of convolutional neural networks in computer vision. *Artif Intell Rev* 57, 99. https://doi.org/10.1007/s10462-024-10721-6

Zhen, L., & Bărbulescu, A. (2024). Comparative analysis of convolutional neural network-long short-term memory, sparrow search algorithm-backpropagation neural network, and particle swarm optimization-extreme learning machine models for the water discharge of the Buzău River, Romania. *Water*, 16(2), 289. https://doi.org/10.3390/w16020289