2024 World Journal of Innovation and Modern Technology, Vol.7, Issue 2 (Apr)
ISSN 2682-5910
DOI: 10.53469/wjimt.2024.07(02).03

A Novel Approach to Credit Card Security with Generative Adversarial Networks and Security Assessment

Han Wang[1,*], Qiaozhi Bao[2], Zuwei Shui[3], Lianwei Li[4], Huan Ji[5]

[1]Financial Mathematics,University of Southern California, LA, USA
[2]Statistics,North Carolina State University, NC, USA
[3]Information Studies,Trine University, Phoenix, USA
[4]Computer Science,University of Texas at Arlington, State of Texas, USA
[5]Engineering Management,Trine University, Phoenix, AZ, USA
Corresponding Author: aimeewang0028@gmail.com

Abstract: ***With the rapid development of Internet technology, many industries have begun digital transformation. However, while bringing convenience to users, the Internet has also become a hotbed for criminals to commit fraud. On the one hand, a large number of users on the Internet more or less left data, criminals can use this information to practice accurate fraud users, improve the success rate of fraud; On the other hand, online financial transactions such as banks and e-commerce also provide more ways for criminals to commit fraud. Therefore, all kinds of fraud methods emerge in an endless flow, through the telephone, information, fishing and other means of fraud, not only to bring hundreds of millions of losses to the society every year, but also to the safety of people's lives have a huge threat. Monitoring and preventing online fraud is an important part of the cybersecurity industry. For known network fraud, based on the phishing site domain name, the account number and mobile phone number that send fraudulent information, simple and effective supervision and defense can be carried out through the blacklist. However, it is difficult for traditional means to effectively defend against undocumented fraud. With the development of machine learning technology, it is the main research direction of fraud detection methods to discover the information sources and characteristics of the information content through machine learning technology, and make real-time and continuous accurate judgments. This paper realizes credit card fraud detection by generating adversarial network technology, so as to prevent network security risks.***

Keywords: Network security; GANs; Credit card fraud; Machine learning

1. INTRODUCTION

Artificial intelligence will be the ultimate Google. The ultimate search engine that understands everything on the web. It will understand exactly what you want, and it will give you the right thing. We are far from that now. But we can gradually get closer to that point, and that's basically our job." -- Larry Page, co-founder and developer of Google.

Because ML algorithms are able to learn from historical fraud patterns and identify them in future transactions, fraud detection using machine learning becomes possible. In terms of information processing speed, machine learning algorithms appear to be more efficient than humans. In addition, machine learning algorithms are able to detect complex fraud features that humans simply cannot detect[1-3]. A rules-based fraud prevention system means creating precise written rules that "tell" the algorithm which types of operations look normal and should be allowed, and which shouldn't because they look suspicious. However, writing rules takes a lot of time. Moreover, manual interactions in the e-commerce world are so dynamic that things can change significantly in a matter of days. Here, machine learning fraud detection methods will come in handy to learn new patterns. ML methods show better performance as the data sets they fit grow - meaning that the more samples of fraudulent operations they accept, the better their ability to identify fraud. The principle does not apply to rules-based systems, as long as they never evolve on their own. In addition, data science teams should be aware of the risks associated with rapid model scaling; If the model does not detect fraud and incorrectly flags it, this will lead to underreporting in the future.

Payment fraud detection is the most common type of fraud solved by artificial intelligence (AI). It is as varied as the fraudsters can imagine[4]. However, here are some of the most common types of payment fraud: lost card, stolen card, fake card, stolen card ID, and card not received. The recent emergence of cards with chips (EMV cards) has helped reduce card fraud in Europe, but not in the United States, where the elimination process for magnetic stripe cards has been very slow. Cardless transactions come in many forms. After attacking the user by phishing,

contacting his or her mobile provider, and breaking into the account online in a way that allows the criminals to gather enough card details, the fraudster orders goods or loans. A loan scam may occur if someone contacts you to offer a loan on unrealistically good terms, the lender does not provide a check confirming the loan, the lender asks for bank details or an advance payment, or the company pretends to be from a certain country but the number is international[5-7]. This fraudster can also make illegal charges through application fraud, which means they apply for a card in your name by filling out stolen information. After obtaining confidential information in different ways online, they can call the credit card company, pretend to be the cardholder, say they need a new credit card, and send it to an address. If the address is stolen, you can change the address in your account.

To sum up, this paper conducts more in-depth research and discussion on credit card fraud, and analyzes how to prevent and detect credit card fraud in advance and ensure network security through machine learning GANs method in the field of artificial intelligence.

2. RELATED WORK

Information about credit cards and online payments gives fraudsters the opportunity to illegally use this information for profit. Looking back at the 2022 data, IC3 received 800,944 network complaints, which is 5% less than in 2021. However, total losses grew from $6.9 billion in 2021 to more than $10.2 billion in 2022. Based on data from the past five years, IC3 receives an average of about 652,000 complaints per year. These complaints relate to a variety of Internet scams, affecting victims worldwide. Credit card fraud is the most common type of payment fraud because digitally stored details give criminals a higher chance of getting away with it. Moreover, transactions are harder to verify[8]. Online shopping is often the first step for credit card thieves, as this requires the criminal to simply enter the credit card information into the necessary fields; Not all stores require additional verification. Criminals can also sell credit card information to other criminals for as little as $45.

2.1 Machine learning fraud detection model

(1) Email phishing detection model

Phishing emails represent spam with fraudulent intent. Phishers make fake websites and their urls highly similar visually and semantically to the original website. They are mainly threats to banking, multinational corporations and even medical institutions. Logistic regression is one of the classic machine learning algorithms for phishing detection. Logistic regression uses a linear model to predict numbers in the range of "0" or "1", which means spam or not.

Another approach is to extract features from websites and classify them as false or not using traditional machine learning classification models such as SVM, Naive Bayes, and extreme learning machines. The first stage before classification involves NLP processing the text from the website and providing semantic analysis of the text.

Typically, phishing detection is addressed as a supervised machine learning problem that involves collecting a large number of fake emails with fake urls from the original source along with an equal number of legitimate emails and websites to train the model[9-10]. The most obvious features that help classify an email as "phishing" are: the use of the "at" symbol in the URL address so that the browser cannot read the symbol before the "at", or that the address bar was downloaded from a domain other than the domain displayed in the address bar. In addition, the length of a site's registration can identify whether the site is a fake, because unlike phishing sites, trusted resources are likely to register their domain for a long time.

(2) Credit card fraud detection model

Fraud models can be solved by supervised and unsupervised machine learning algorithms. In the first case, a traditional classification algorithm is used; In the second case, we can use anomaly detection techniques. The use of neural networks is also effective, but it requires a lot of training data, with two types of data points in equal numbers: abnormal and normal. However, in the case of fraud detection, there is always a lack of balanced data sets.

(3) Identity document forgery detection model

Identity document forgery detection first involves image processing. Some techniques are used to understand the visual information carried by images[11]. CNN models are typically trained to perform this task, while neural networks are built in a way designed to minimize losses. The CNN mimics the work of the human visual cortex - the part of the brain responsible for processing visual information. Just as supervised learning requires collecting a set of fake and real document images, the dataset needs to have a sufficient number of photos from both categories. Configuring neural networks to perform at maximum efficiency includes testing different architecture types with different layers and convolutional layer filter sizes. Typically, a convolutional architecture has four convolutional layers. The method has an accuracy of about 98% for detecting ink mismatches in forged documents with blue ink and 88% with black ink.

This forgery detection technique relies on HSI, which stands for Hyperspectral Image Analysis. This method means building an electromagnetic spectrogram to obtain the spectrum of each pixel in the image.

2.2 Generative adversarial Networks (GANs)

Generative adversarial Networks (GANs) are a powerful generative model that can synthesize new realistic images. Through the complete implementation process, readers will gain a deep understanding of how GANs works behind the scenes. This tutorial begins by importing the necessary libraries and loading the Fashion-MNIST dataset that will be used to train Gans. Then, code examples for building the core components of a GAN (generator and discriminator models) are provided. The following sections explain how to build a composite model that trains the generator to fool the discriminator, and how to design a training function to optimize the adversarial process[12-14].

The relationship between the generator and the discriminator is much like the relationship between the counterfeiter and the Appraiser. Counterfeiters continue to create fake goods in order to fool appraisers, and in the process their ability to fake is increasing. Appraisers are constantly testing fakes in order to spot the counterfeiters, and in the process their ability to identify them is getting better and better. Gans belong to counterfeiters, they belong to evaluators, but ultimately they belong to counterfeiters. The ultimate goal of Gans is to train a "perfect" counterfeiter, that is, to produce a product that confuses all cognoscenti.

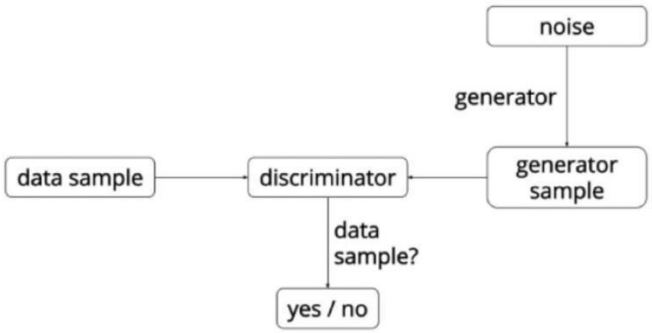

Figure 1: The adversarial network implementation model is generated

In fact, the generation network G receives a random noise z, and generates a picture G(z) through this noise, while the discriminator D needs to judge whether the input picture x is real or not, and the discriminator will output the probability D(x) of judging the picture, D(x)=1 if judged to be a real picture, D(x)=0 if judged to be an untrue picture. In the process of iterative training, the generator model tries its best to generate real pictures to deceive the discriminator, and the discriminator tries its best to distinguish the generator from the real pictures, thus forming a dynamic game process. In the best case, the generator G is enough to be true, and it is difficult for the discriminator to judge the true and false images generated by the generator G. So D(G(z))=0.5.

Its objective function formula is as follows:

$$V(D,G) := \mathbb{E}_{\mathbf{x}\sim\mu}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim\gamma}[\log(1 - D(G(\mathbf{z})))] \quad (1)$$

Where E represents the expected value with respect to the distribution specified in the subscript. The description of the minimax solved by GAN is as follows:

$$\min_G \max_D V(D,G) := \min_G \max_D (\mathbb{E}_{\mathbf{x}\sim\mu}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim\gamma}[\log(1 - D(G(\mathbf{z})))] \quad (2)$$

For a given generator G, maxpV(D,G) optimizes the discriminator D to distinguish between the generated sample G(z) by trying to assign high values to the real sample from the distribution μ and low values to the generated sample (z).

Instead, for a given discriminator D, minG V(D,G) optimizes G such that the resulting sample G(z) will try to "fool" the discriminator D to assign high values. Let y= G(z)€R ", whose distribution is v:=γoG-1, V(D,G) can be rewritten with D and V as:

$$\begin{aligned} \tilde{V}(D,\nu) := V(D,G) &= \mathbb{E}_{\mathbf{x}\sim\mu}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim\gamma}[\log(1 - D(G(\mathbf{z})))] \\ &= \mathbb{E}_{\mathbf{x}\sim\mu}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{y}\sim\nu}[\log(1 - D(\mathbf{y}))] \\ &= \int_{\mathbb{R}^n} \log D(x) d\mu(x) + \int_{\mathbb{R}^n} \log(1 - D(y)) d\nu(y) \end{aligned} \quad (3)$$

For the discriminator, the discriminator should output 0 if it gets a generated picture, and 1 if it gets a real picture, and get the error gradient backpropagation to update the parameters. For the generator, an image is first generated by the generator, and then input to the discriminator to judge and combine the corresponding error gradients, and then backpropagate these image gradients to become the weights that make up the generator[15-17]. Intuitively, the discriminator has to tell the generator how to adjust to make the picture it produces more realistic.

Therefore, in the prevention of credit card fraud detection, the steps of GAN for anomaly detection are outlined:

1. Normal data training:

Gans are trained using a dataset of normal or typical instances of the data (e.g., normal images, normal sensor readings, etc.)[18][19][20]. The generator learns to generate synthetic samples that simulate a normal data distribution, and the discriminator is trained to distinguish between real data and synthetic data.

2. Generation of synthetic data:

Use the trained generator to generate a combined data sample. These synthetic samples should be similar to normal instances in the training data, but we don't need this part of the model.

3. Anomaly detection:

The synthetic data generated by GAN is combined with the original normal data. Use traditional anomaly detection techniques or simple threshold methods to identify instances that deviate significantly from the expected distribution. Instances that do not resemble either the real or synthetic data are considered potential anomalies. (This is an easy way)

4, discriminator as an anomaly detector:

The discriminator was repurposing as an anomaly detector. It is applied to both real and synthetic data in the anomaly detection phase. Instances classified by the discriminator as real may be considered normal, while instances classified as synthetic may be flagged as potential exceptions. (This is an exception detection method using a discriminator alone)

3. METHODOLOGY

3.1 Import data

The data we will use is the Kaggle credit card fraud detection dataset [1]. It contains features V1 through V28, which are the main components obtained by PCA, and ignores time features that are not useful for building the model. The remaining features are the "amount" feature,[21] which contains the total amount of the transaction, and the "category" feature, which contains whether the transaction is a case of fraud.

Import the data by using the 'pd.read_csv' method and view some sample data:

The Convolutional Neural Network (CNN) model is a powerful tool for extracting hierarchical features from structured data. In the context of this experiment, the CNN is utilized to predict the robot's position based on the sensor data collected from the floor. The network architecture is meticulously designed to leverage spatial correlations present in the data.

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	...	V21	V22	V23	V24	
0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	0.090794	...	-0.018307	0.277838	-0.110474	0.066928	0.
1	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	-0.166974	...	-0.225775	-0.638672	0.101288	-0.339846	0.
2	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	0.207643	...	0.247998	0.771679	0.909412	-0.689281	-0.
3	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	-0.054952	...	-0.108300	0.005274	-0.190321	-1.175575	0.
4	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	0.753074	...	-0.009431	0.798278	-0.137458	0.141267	-0.

Conduct data analysis

```
CASE COUNT
--------------------------------------------
Total number of cases are 284807
Number of Non-fraud cases are 284315
Number of Non-fraud cases are 492
Percentage of fraud cases is 0.17
--------------------------------------------
```

It can be seen that out of the 284,807 sample, there were only 492 cases of fraud, which is only 0.17% of the total sample. So, it can be said that the data we are dealing with is highly unbalanced data that needs to be handled carefully when modeling and evaluating.

Next, you'll use the "describe" method in Python to get a statistical view of the fraudulent and non-fraudulent transaction amount data[22].

In this process, the independent variable (X) and the dependent variable (Y) are defined. The data is divided into training sets and test sets using defined variables for further modeling and evaluation. You can easily split the data using the "train_test_split" algorithm in python.

```
X_train samples :  [[-1.11504743  1.03558276  0.80071244 -1.06039825  0.03262117  0.85342216
  -0.61424348 -3.23116112  1.53994798 -0.81690879 -1.30559201  0.1081772
  -0.85960958 -0.07193421  0.90665563 -1.72092961  0.79785322 -0.0067594
   1.95677806 -0.64489556  3.02038533 -0.53961798  0.03315649 -0.77494577
   0.10586781 -0.43085348  0.22973694 -0.0705913  -0.30145418]]
X_test samples :  [[-0.32333357  1.05745525 -0.04834115 -0.60720431  1.25982115 -0.09176072
   1.1591015  -0.12433461 -0.17463954 -1.64440065 -1.11886302  0.20264731
   1.14596495 -1.80235956 -0.24717793 -0.06094535  0.84660574  0.37945439
   0.84726224  0.18640942 -0.20709827 -0.43389027 -0.26161328 -0.04665061
   0.2115123   0.00829721  0.10849443  0.16113917 -0.19330595]]
y_train samples :  [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
y_test samples :  [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
```

3.2 Build a model

Six different types of classification models need to be built, namely decision trees, K-nearest neighbor (KNN), logistic regression, support vector machines (SVM), random forests, and XGBoost. While there are many other models we could use, we have chosen the most popular ones for solving classification problems. All of these models are easy to build and can be built using algorithms provided by the scikit-learn package. For the XGBoost model only, the xgboost package will be used. In order to simplify the period, only one of the models is shown in this article for the subsequent experiment[23-25].

The core model of this paper is K-cluster model, so in k-nearest neighbor (KNN), "KNeighborsClassifier" algorithm is used to build the model, and "n_neighbors=5" is set. The value of 'n_neighbors' is selected randomly, but can be purposefully selected by iterating over a series of values and then storing the predicted value in the 'knn_yhat' variable after fitting the model.

```
ACCURACY SCORE
----------------------------------------------------------------------
Accuracy score of the Decision Tree model is 0.9993679997191109
----------------------------------------------------------------------
Accuracy score of the KNN model is 0.9995259997893332
----------------------------------------------------------------------
Accuracy score of the Logistic Regression model is 0.9991924440855307
----------------------------------------------------------------------
Accuracy score of the SVM model is 0.9993153330290369
----------------------------------------------------------------------
Accuracy score of the Random Forest Tree model is 0.9993153330290369
----------------------------------------------------------------------
Accuracy score of the XGBoost model is 0.9994908886626171
----------------------------------------------------------------------
```

According to the accuracy score evaluation index, KNN model is the most accurate model, while Logistic regression model is the least accurate. However, when we round up the results of each model, we get 99% accuracy, so it can be seen that this model is more consistent with the experimental model of credit card fraud that implemented this paper.

3.3 K-Nearest Neighbors

```
knn_cm_plot = plot_confusion_matrix(knn_matrix,
                                    classes = ['Non-Default(0)','Default(1)'],
                                    normalize = False, title = 'KNN')
plt.savefig('knn_cm_plot.png')
plt.show()
```

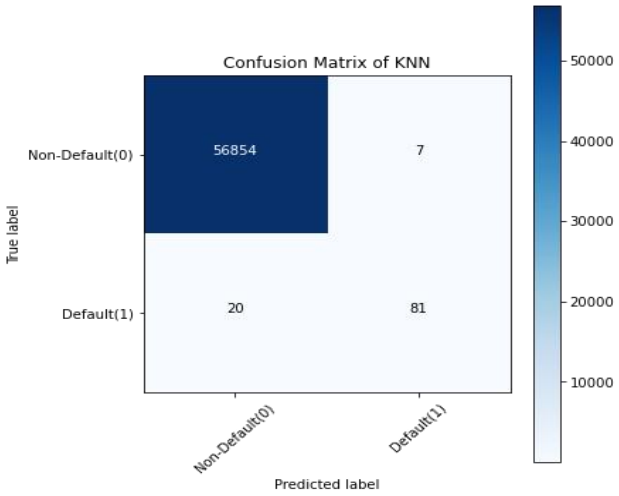

Figure 2: K-Nearest model result

The first line.

The first line is the transaction with an actual fraud value of 0 in the test set. It can be calculated that 56,861 of the fraud values are 0. Of the 56,861 non-fraudulent transactions, the classifier correctly predicted 56,854 of them to be 0 and predicted 7 of them to be 1[26-28]. This means that for 56,854 non-fraudulent transactions, the actual churn value in the test set was 0, which the classifier also correctly predicted. We can say that our model has classified non-fraudulent transactions and transactions are good.

The second line.

It looks like 101 transactions have a fraud value of 1. The classifier correctly predicted 79 of them as 1 and incorrectly predicted 22 of them as 0. The wrong predicted value can be considered an error in the model.

When comparing the confusion matrix of all models, it can be seen that the K-Nearest Neighbors model does a very good job of classifying fraudulent transactions from non-fraudulent transactions, followed by the XGBoost model. Therefore, it can be concluded that the most suitable model for this case is K-Nearest Neighbors.

4. CONCLUSION

In summary, the paper delves into the pressing issue of credit card fraud in the digital age, exacerbated by the rapid expansion of internet technologies. It discusses the various methods employed by criminals to perpetrate fraud online, from phishing to identity theft, leading to significant financial losses and posing serious threats to cybersecurity. Traditional means of fraud detection, though effective to some extent, struggle to combat undocumented fraud effectively.To address this challenge, the paper proposes leveraging machine learning techniques, particularly Generative Adversarial Networks (GANs), for credit card fraud detection. GANs, known for their ability to generate synthetic data resembling real instances, offer a promising approach to identify anomalies and potential fraud cases[29]. By training GANs on normal data and utilizing the discriminator as an anomaly detector, the paper outlines a methodology for detecting fraudulent transactions effectively.

Moreover, the study evaluates various classification models, including decision trees, logistic regression, and K-nearest neighbors, for fraud detection, highlighting the superiority of the K-nearest neighbors model in accurately classifying fraudulent transactions.

Looking ahead, the application of Generative Adversarial Networks (GANs) in fraud detection holds great potential for the future of cybersecurity. As technology continues to evolve, GANs could be further refined and integrated into advanced fraud detection systems, enabling real-time and continuous monitoring of transactions. Additionally, with ongoing research and development, GANs could enhance their capabilities to detect even subtle anomalies, thereby fortifying cybersecurity measures against increasingly sophisticated fraud schemes in the digital realm. The future of fraud detection lies in harnessing the power of machine learning and artificial intelligence to stay one step ahead of cybercriminals, ensuring the safety and security of online transactions for individuals and businesses alike.

ACKNOWLEDGEMENT

I would like to thank Professor Le Yang for his outstanding contributions and professional insights in the field of credit card security. His paper, The Credit Card Anti-fraud Detection Model in the Context of Dynamic Integration Selection Algorithm provides a profound theoretical basis and method guidance for the research of this paper. The discussion of credit card fraud detection model in this paper not only enrichis my understanding of credit card security, but also provides valuable inspiration and direction for my research. Through his in-depth research, Professor Le Yang has made outstanding contributions to the development of the credit card security field and provided useful ideas and solutions for the industry.

In addition, I would like to thank Professor Le Yang for his guidance and support in academic exchanges and research cooperation. He not only provided me with professional advice and technical support, but also shared his research results in academic forums and seminars, promoting knowledge exchange and cooperation in our field. With the help of Professor Le Yang, I was able to deeply explore the cutting-edge issues in the field of credit card security, and put forward the innovative ideas and methods in this paper.

Finally, I would like to express my sincere thanks and high respect to Professor Le Yang once again. Thank you for your selfless sharing and professional support, and look forward to more academic exchanges and cooperation opportunities in the future to jointly promote the development of credit card security.

REFERENCES

[1] Chen, Wangmei, et al. “Applying Machine Learning Algorithm to Optimize Personalized Education Recommendation System”. Journal of Theory and Practice of Engineering Science, vol. 4, no. 01, Feb. 2024, pp. 101-8, doi:10.53469/jtpes.2024.04(01).14.

[2] Dong, Xinqi, et al. “The Prediction Trend of Enterprise Financial Risk Based on Machine Learning ARIMA Model”. Journal of Theory and Practice of Engineering Science, vol. 4, no. 01, Jan. 2024, pp. 65-71, doi:10.53469/jtpes.2024.04(01).09.

[3] “A Deep Learning-Based Algorithm for Crop Disease Identification Positioning Using Computer Vision”. International Journal of Computer Science and Information Technology, vol. 1, no. 1, Dec. 2023, pp. 85-92, https://doi.org/10.62051/ijcsit.v1n1.12.

[4] Du, S., Li, L., Wang, Y., Liu, Y., & Pan, Y. (2023). Application of HPV-16 in Liquid-Based thin Layer Cytology of Host Genetic Lesions Based on AI Diagnostic Technology Presentation of Liquid. Journal of Theory and Practice of Engineering Science, 3(12), 1-6.
[5] Xin, Q., He, Y., Pan, Y., Wang, Y., & Du, S. (2023). The implementation of an AI-driven advertising push system based on a NLP algorithm. International Journal of Computer Science and Information Technology, 1(1), 30-37.
[6] He, Yuhang, et al. "Intelligent Fault Analysis With AIOps Technology". Journal of Theory and Practice of Engineering Science, vol. 4, no. 01, Feb. 2024, pp. 94-100, doi:10.53469/jtpes.2024.04(01).13.
[7] Chen , J., Xiong, J., Wang, Y., Xin, Q., & Zhou, H. (2024). Implementation of an AI-based MRD Evaluation and Prediction Model for Multiple Myeloma. Frontiers in Computing and Intelligent Systems, 6(3), 127-131. https://doi.org/10.54097/zJ4MnbWW
[8] Tan, Kai, et al. "Integrating Advanced Computer Vision and AI Algorithms for Autonomous Driving Systems". Journal of Theory and Practice of Engineering Science, vol. 4, no. 01, Jan. 2024, pp. 41-48, doi:10.53469/jtpes.2024.04(01).06.
[9] "Exploring New Frontiers of Deep Learning in Legal Practice: A Case Study of Large Language Models". International Journal of Computer Science and Information Technology, vol. 1, no. 1, Dec. 2023, pp. 131-8, https://doi.org/10.62051/ijcsit.v1n1.18.
[10] Development of Machine Learning and Artificial Intelligence in Toxic Pathology. (2024). Frontiers in Computing and Intelligent Systems, 6(3), 137-141. https://doi.org/10.54097/Be1ExjZa
[11] Jili Qian, et al. "Analysis and Diagnosis of Hemolytic Specimens by AU5800 Biochemical Analyzer Combined With AI Technology". Frontiers in Computing and Intelligent Systems, vol. 6, no. 3, Jan. 2024, pp. 100-3, https://doi.org/10.54097/qoseeQ5N.
[12] Pan, Yiming, et al. "Application of Three-Dimensional Coding Network in Screening and Diagnosis of Cervical Precancerous Lesions". Frontiers in Computing and Intelligent Systems, vol. 6, no. 3, Jan. 2024, pp. 61-64, https://doi.org/10.54097/mi3VM0yB.
[13] Wei, Kuo, et al. "Strategic Application of AI Intelligent Algorithm in Network Threat Detection and Defense". Journal of Theory and Practice of Engineering Science, vol. 4, no. 01, Jan. 2024, pp. 49-57, doi:10.53469/jtpes.2024.04(01).07.
[14] "Unveiling the Future Navigating Next-Generation AI Frontiers and Innovations in Application". International Journal of Computer Science and Information Technology, vol. 1, no. 1, Dec. 2023, pp. 147-56, https://doi.org/10.62051/ijcsit.v1n1.20.
[15] Zong, Yanqi, et al. "Improvements and Challenges in StarCraft II Macro-Management A Study on the MSC Dataset". Journal of Theory and Practice of Engineering Science, vol. 3, no. 12, Dec. 2023, pp. 29-35, doi:10.53469/jtpes.2023.03(12).05.
[16] An Overview of the Development of Stereotactic Body Radiation Therapy. (2024). Frontiers in Computing and Intelligent Systems, 6(3), 56-60. https://doi.org/10.54097/09nIy12x.
[17] Zheng, Jiajian, et al. "The Credit Card Anti-Fraud Detection Model in the Context of Dynamic Integration Selection Algorithm". Frontiers in Computing and Intelligent Systems, vol. 6, no. 3, Jan. 2024, pp. 119-22, https://doi.org/10.54097/a5jafgdv.
[18] Wang, Sihao, et al. "Diabetes Risk Analysis Based on Machine Learning LASSO Regression Model". Journal of Theory and Practice of Engineering Science, vol. 4, no. 01, Jan. 2024, pp. 58-64, doi:10.53469/jtpes.2024.04(01).08.
[19] Gao, Longsen, et al. "Autonomous multi-robot servicing for spacecraft operation extension." 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2023.
[20] "Enhancing Computer Digital Signal Processing through the Utilization of RNN Sequence Algorithms". International Journal of Computer Science and Information Technology, vol. 1, no. 1, Dec. 2023, pp. 60-68, https://doi.org/10.62051/ijcsit.v1n1.09.
[21] Yu, L., Liu, B., Lin, Q., Zhao, X., & Che, C. (2024). Semantic Similarity Matching for Patent Documents Using Ensemble BERT-related Model and Novel Text Processing Method. arXiv preprint arXiv:2401.06782.
[22] Huang, J., Zhao, X., Che, C., Lin, Q., & Liu, B. (2024). Enhancing Essay Scoring with Adversarial Weights Perturbation and Metric-specific AttentionPooling. arXiv preprint arXiv:2401.05433.
[23] Pan, Yiming, et al. "Application of Three-Dimensional Coding Network in Screening and Diagnosis of Cervical Precancerous Lesions". Frontiers in Computing and Intelligent Systems, vol. 6, no. 3, Jan. 2024, pp. 61-64, https://doi.org/10.54097/mi3VM0yB.
[24] K. Jin, Z. Z. Zhong and E. Y. Zhao, "Sustainable Digital Marketing Under Big Data: An AI Random Forest Model Approach," in IEEE Transactions on Engineering Management, vol. 71, pp. 3566-3579, 2024, doi: 10.1109/TEM.2023.3348991.

[25] Zhang, Yufeng, et al. "Manipulator Control System Based on Machine Vision." International Conference on Applications and Techniques in Cyber Intelligence ATCI 2019: Applications and Techniques in Cyber Intelligence 7. Springer International Publishing, 2020.
[26] “Jin, Keyan. "Impacts of Word of Mouth (WOM) on E-Business Online Pricing." JGIM vol.31, no.3 2023: pp.1-17. http://doi.org/10.4018/JGIM.324813
[27] “Machine Learning Model Training and Practice: A Study on Constructing a Novel Drug Detection System”. International Journal of Computer Science and Information Technology, vol. 1, no. 1, Dec. 2023, pp. 139-46, https://doi.org/10.62051/ijcsit.v1n1.19.
[28] Q. Cheng, M. Tian, L. Yang, J. Zheng, and D. Xin, “Enhancing High-Frequency Trading Strategies with Edge Computing and Deep Learning”, Journal of Industrial Engineering & Applied Science, vol. 2, no. 1, pp. 32–38, Feb. 2024.
[29] Cai, J., Ou, Y., Li, X., Wang, H. (2021). ST-NAS: Efficient Optimization of Joint Neural Architecture and Hyperparameter. In: Mantoro, T., Lee, M., Ayu, M.A., Wong, K.W., Hidayanto, A.N. (eds) Neural Information Processing. ICONIP 2021. Communications in Computer and Information Science, vol 1516. Springer, Cham. https://doi.org/10.1007/978-3-030-92307-5_32
[30] Yimin Ou, Rui Yang, Lufan Ma, Yong Liu, Jiangpeng Yan, Shang Xu, Chengjie Wang, Xiu Li,UniInst: Unique representation for end-to-end instance segmentation,Neurocomputing,Volume 514,2022,Pages 551-562,ISSN 0925-2312,https://doi.org/10.1016/j.neucom.2022.09.112.