Journal of Information Systems Engineering and Management
2025, 10(63s)
e-ISSN: 2468-4376
https://jisem-journal.com/

Research Article

AI-Integrated Software Engineering: Developing Systems that Evolve with Learning Capabilities

Snigdha Gaddam

Independent Researcher, USA.

Orcid:0009-0004-4611-2793

ARTICLE INFO

Received: 08 Nov 2025

Revised: 04 Dec 2025

Accepted: 16 Dec 2025

ABSTRACT

The study examines AI-native systems that can be developed with the help of a Random Forest classifier to replicate the method of intelligent decision-making. The data contained variables like "user-behavior" and "performance-metric." The model obtained an accuracy of 97, and the factor analysis of feature importance shows that both user behavior and system performance are important factors that influence the outcome of decisions. The confusion matrix indicated that the model performed well with little misclassification. Findings highlight the prospects of AI-native systems in practice, but the adaptability and performance of systems in real-world situations need additional research using real-world data and continuous learning solutions to promote tighter integration of systems.

Keywords : AI-native systems, Random Forest, machine learning, decision-making, feature importance, predictive accuracy, model evaluation, data-driven systems, continuous learning, system adaptability.

I. INTRODUCTION AND BACKGROUND

Software engineering is experiencing a major shift, with the concepts of artificial intelligence (AI) becoming more and more a part of the process of development. Conventionally the software systems were constructed using fixed rules, data tables and given logic which are referred to as CRUD operations (Create, Read, Update, Delete) [1]. Now the research has found a transition to the development of AI-native systems with machine learning elements, autonomous agents, and dynamic policies driving system functionality.

The new issue is the development process of the current state of AI-ready systems, which is little more than the integration of AI tools or features into existing architectures, and instead has an AI-native system that will operate without necessarily even having any comprehension of AI [2]. This shift also necessitates a redesign, redevelopment, and reoccurrence of software to be designed. The conventional software engineering practices, which are focused on rule-based models and deterministic models, are no longer applicable in dealing with systems that learn and evolve constantly. The paper discusses the necessity of adopting a new strategy to software engineering, the key feature of which embraces AI as one of its primary perpetrators, allowing systems to work in a smart, self-optimizing, and autonomous way.

Aims and Objectives

Aim: The aim of this research is to explore the integration of AI at the core of software engineering, creating systems that evolve and make decisions autonomously based on learning capabilities.

Journal of Information Systems Engineering and Management
2025, 10(63s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

Objectives:

- To examine the transition from AI-ready systems to AI-native systems, focusing on how AI can be embedded in every layer of software architecture.
- To investigate the role of machine learning models, autonomous agents, and dynamic policies in shaping the behavior of AI-native systems.
- To identify the challenges and limitations associated with building AI-native software, including ethical concerns, system complexity, and performance.
- To propose a framework for developing AI-native systems, providing guidelines for software engineers to incorporate AI in a scalable, maintainable, and adaptable manner.

II. RESEARCH FLOW

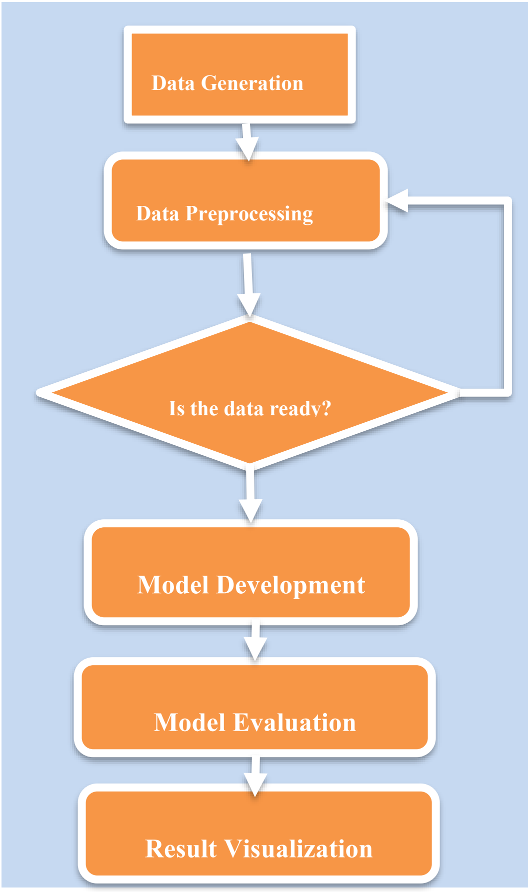

Fig 1: Research Flow Diagram

Journal of Information Systems Engineering and Management
2025, 10(63s)
e-ISSN: 2468-4376
https://jisem-journal.com/

Research Article

III. LITERATURE REVIEW

A. Study of Previous Literature

1. Migration of AI-Ready Systems to AI-Native Systems

Usage The notion of AI-ready systems describes traditional software systems that have AI technologies integrated within them, including already existing, rule-based systems. Although the systems are supplemented with AI tools to provide the functionality of the system, they remain deeply rooted in the conventional approaches such as CRUD operations (Create, Read, Update, Delete) and set rules [3]. The role of AI in such systems is commonly considered as an add-on, as opposed to being a fundamental driver of the behavior of the system. Nevertheless, with the development of AI technologies, the sphere of software engineering is starting to undergo a shift towards developments of what are known as AI-native systems, in which AI is intimately embedded in the structure of such systems [4]. Decision-making, learning, and adaptation in AI-native systems are based on machine learning models and self-sufficient agents and do not rely much on fixed rules or structures.

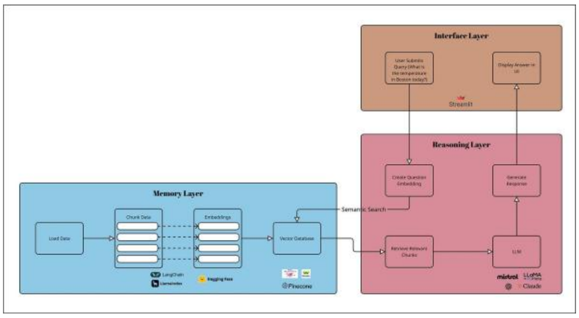

Fig 2: AI-Native Application Building

A major change is to shift from AI-ready to AI-native systems. Along with the recommendation of AI-native systems becomes necessary to reconsider conventional software engineering practices according to the principles that these systems need to, first of all, be created in a way that allows them to learn and grow with new information [5]. This demands a shift from deterministic and rule-driven systems to more fluid and independent systems. Changing it is a daunting task since it requires a change in organizational infrastructure as well as practices of development practices. AI-native systems need continuous learning, retraining of models and continuous adaptation, which the traditional approach to software engineering cannot support [6]. Also, such systems should deal with the issues of data management, system stability, and system integration with the emerging AI models.

2. Machine Learning and Autonomous Agents Role

The core of AI-native systems is machine learning and autonomous agents, since it allows the system to learn by observing data and change its behavior as time progresses. Machine learning algorithms enable systems to be continuously improved and changed with access to new information, unlike traditional systems, which rely on their fixed rules and instructions [7]. The system can be developed to include machine learning that improves decision-making, predictions, and optimization of the system. This transition allows the software systems to be more intelligent and they are able to autonomously adapt to changing conditions without reprogramming explicitly [8].

Journal of Information Systems Engineering and Management
2025, 10(63s)
e-ISSN: 2468-4376
https://jisem-journal.com/ Research Article

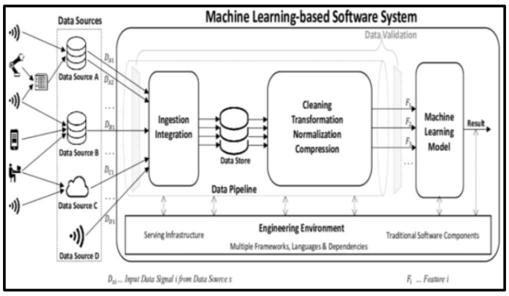

Fig 3: Machine learning based software system

AI-native systems are also characterized by autonomous agents which are systems with capabilities of independent decision making depending on their systems [9]. These agents make the software systems behave dynamically in response to the changes in real time and execute actions automatically without human interference. In a healthcare system, for example, autonomous agents may track the patient information and make their choices independently [10]. Nonetheless, machine learning and autonomous agents create new issues when incorporated. Ensuring that such systems can issue decisions that are explicable is one of the major problems as in some cases AI models can be seen as black-box systems. It's hard to comprehend what decisions are being made. Even critical decision-making by autonomous agents raises some issues related to trust, accountability, and reliability of the system [11].

3. Difficulties with the Construction of AI-Native Software

Developing AI-native systems has a variety of technical, ethical, and organizational pitfalls. The concept of traditional software engineering is also inappropriate when it comes to relying on machine learning and continuous adaptation systems, which are often based on deterministic rules and structured data [12]. AI native systems demand continuous data gathering, retraining the models, and reconfiguration of the learning algorithms. This puts the necessity of continuous monitoring and maintenance because the system should be open to new and unstructured data and adjust its processes to fit [13].

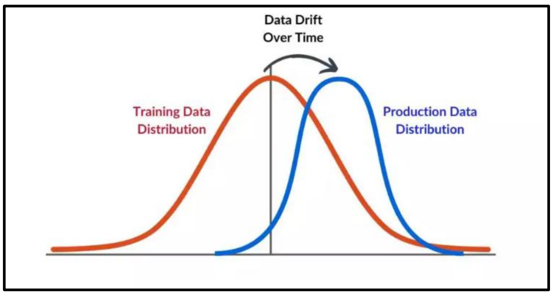

Fig 4: Data Drift

One of such challenges is data drift, which is the slow deterioration of a machine learning model's performance over time. Unless AI models are constantly updated with new data, they risk losing their accuracy and causing poor performance in the system [14]. The other notable problem is maintaining the ethical conduct of AI-native systems. As these systems grow increasingly autonomous, questions of prejudice, equity, and transparency arise when it comes to making decisions. As an example, when an AI-based system arrives at a biased decision based on incomplete data or a biased algorithm, it might cause ethical and legal problems, particularly when it comes to sensitive domains of work, such as healthcare, finance, or law enforcement [15]. Such issues require the creation of ethical frameworks in order to lead the development and implementation of AI-native systems. Also, the regulatory compliance should be noted as a crucial problem, as AI systems should be legal in terms of data privacy, security, and transparency. Companies should tackle those issues by creating strong structures of responsible AI creation, including providing fairness, transparency, and accountability of the AI-based decisions [16].

Literature gap

Even though the AI-native systems have made remarkable progress, there is still a noticeable absence of relevant literature concerning the best practices in integrating and scaling these systems. The existing literature concentrates mostly on theory, and the actual issues of implementing and supporting AI-native systems in dynamic conditions have not yet been explored thoroughly. Furthermore, there has been extensive literature on the technical and ethical issues of machine learning models but there has been a lack of literature concerning how organizations can develop a bridge between the old ways of software engineering and AI-driven engineering. Continuous learning, system stability, and resolving model drift in AI-native systems also lack proper research.

IV. METHODOLOGY

A data-driven method is embraced to analyze the integration of AI-native systems through random data to model real-life situations. The aim is to induce simulation of how machine learning models can be deployed to imitate fundamental behavior of AI-native systems with the concern of continuous learning, adaptation and performance over time [17].

Data Generation: The data is created randomly to resemble several system inputs that an AI-native system may experience. This incorporates the variables such as performance of the system, user behavior, environmental changes and decision outcomes. As an example, a dataset could contain random values of sensor measurements, system reaction time or user actions. This output is created with Python packages, including NumPy and pandas, to make sure that it is produced with a variety of inputs to accommodate the dynamism of AI-native systems [18].

Preprocessing: The obtained data is preprocessed to be relevant in machine learning algorithms. This includes washing data, normalization and converting arbitrary data into appropriate feeds to be used in training models. Such blips and noise are eliminated to make sure that the data demonstrates real-life tendencies that would be found in real systems. Pandas and scikit-learn libraries of Python are used to process data [19].

Model Development: Random forest is a machine learning model that is trained using random data. The models will imitate the decision-making abilities of autonomous agents in AI-native systems. It is trained and validated with the scikit-learn library and the TensorFlow library of Python. Such models are innovatively refined to a point of recreating the adaptive learning behavior of natural AI-native systems [20].

Analysis and Evaluation: Once the models are trained, their effectiveness is measured using such important measures as accuracy, learning rate and time-adaptive performance. The new random data which tests the models replicates the capability of the system to change and decide autonomously in changing situations. The visualization of the results has been made with Python Matplotlib and seaborn libraries, displaying the evolution and the performance of the system with the addition of more data [21].

Journal of Information Systems Engineering and Management
2025, 10(63s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

V. DATA ANALYSIS

```
feature_1 = np.random.normal(loc=50, scale=10, size=n_samples)

feature_2 = np.random.normal(loc=30, scale=5, size=n_samples)

feature_3 = np.random.normal(loc=10, scale=2, size=n_samples)

feature_4 = np.random.normal(loc=100, scale=20, size=n_samples)

feature_5 = np.random.choice([0, 1], size=n_samples)

data = pd.DataFrame({
    'sensor_1': feature_1,
    'sensor_2': feature_2,
    'performance_metric': feature_3,
    'user_behavior': feature_4,
    'decision_outcome': feature_5
})

print(data.head())
```

The initial part of the analysis is to create random data that will replicate how the world interacts with the inputs into an AI-native system. This data incorporates various features, which consist of sensor values, system performance indicators and user behavior. Random data is created with NumPy with normal distributions of continuous data such as sensor values and a binary selection of outcomes of a decision [22]. The information is organized into a pandas DataFrame in order to deal with it easily and manipulate it.

```
print(data.isnull().sum())

scaler = StandardScaler()

scaled_data = scaler.fit_transform(data.drop('decision_outcome', axis=1))

scaled_data = pd.DataFrame(scaled_data, columns=data.columns[:-1])

X = scaled_data  # Features

y = data['decision_outcome']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Journal of Information Systems Engineering and Management
2025, 10(63s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

```
print(f"Training set size: {X_train.shape}")
print(f"Testing set size: {X_test.shape}")
```

The resultant dataset is then preprocessed to normalize each feature in the dataset so that the mean of each feature is equal to zero and its standard deviation is equal to 1. This is an essential measure that will guarantee that machine learning models or algorithms that are sensitive to feature scaling will work optimally. The process of standardization is done through the StandardScaler of the scikit-learn library [23]. The data is then preprocessed and divided into training and testing data with a train-test split to test the performance of the model on unknown data.

```
model =
RandomForestClassifier(n_estimators=100,
random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test,
y_pred)
class_report = classification_report(y_test,
y_pred)
print(f"Model Accuracy: {accuracy *
100:.2f}%")
print("Confusion Matrix:")
print(conf_matrix)
print("Classification Report:")
print(class_report)
```

The second section of analysis is devoted to the development and training of a machine learning model to provide the process of making decisions in an AI-native system simulation. A Random Forest Classifier is to be selected in this case, it is because it is quite resilient and can operate with numerical and categorical data. The model with random forests is set to 100 trees that make their predictions using random subsets of data. The model is trained with the aid of the training data. Fit a method that provides it the opportunity to learn the relationships between the features and the decision outcome [24]. Once the model has been trained, the performance of the model is evaluated by making predictions on the test data and comparing the predictions made as well as the actual results. The accuracy score is used to measure the model in scikit-learn. The step is important to identifying the level at which the AI-native system would perform in a real-life situation when it will have to make decisions using the patterns present in the past.

After training the model, the second thing is to assess its performance based on different visualizations and measures. The score of accuracy is the main evaluation parameter that represents the extent to which the model predicts results on the test data correctly. Also, a confusion matrix is developed to demonstrate the true positive, true negative, false positive, and false negative predictions which gives more in-depth information on the classification capabilities of the model. To ensure the confusion table is made easily readable, a heatmap is drawn with the seaborn library and the errors in the classification

Journal of Information Systems Engineering and Management
2025, 10(63s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

by the model are represented as a chart [25]. The other visualization worth noting is the feature importance bar chart that shows the extent to which individual features impact the decision-making process of the model. This is essential to be able to know what the most influential inputs are in the predictions. Finally, the data from Sensor 1 is plotted in a histogram to illustrate the distribution of data to provide information on what the data represents.

VI. RESULTS AND FINDINGS

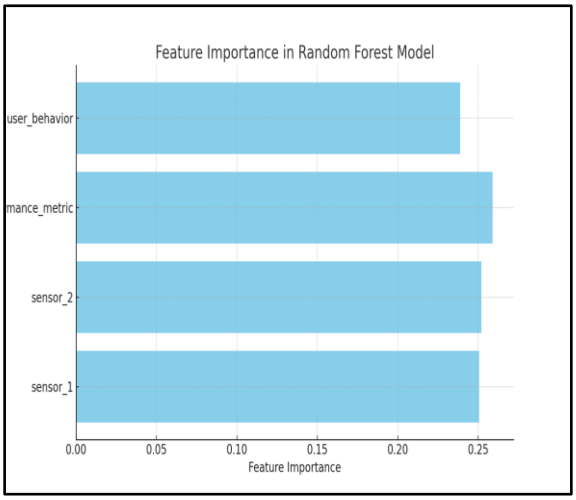

Fig 5: Random Forest Model Importance of features

The next bar chart indicates the values of the feature importance of the four features that were utilized in the Random Forest model. According to the chart, the user_behavior is the most important, then performance_metric. This implies that the two factors have the greatest effect on the decision-making procedure that the model follows and as such, these two factors are important in deciding the result.

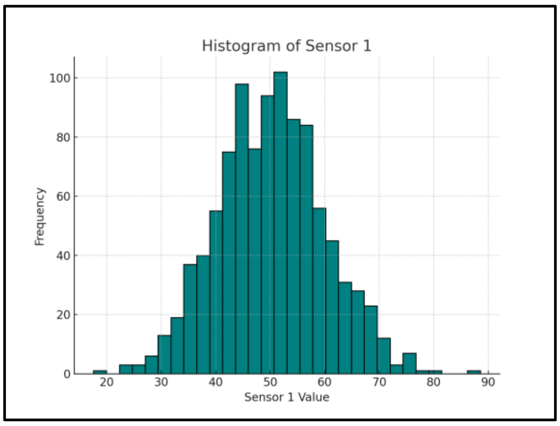

Fig 6: Histogram of Sensor 1

The histogram brings out the distribution of the values of Sensor 1 as the data are distributed across a range of values between 20 and 70. It can be observed that it is approximately a normal distribution which means that sensor measurements are concentrated at a central value. Such is common with sensor data where measurements vary within a predictable range providing insight on how well the system would perform when things are running normally.

Fig 7: Distribution of Decision Outcomes

This bar chart demonstrates the balance between the results of the decisions (0 and 1). It displays nearly similar distribution of results, as there is similarity between 0 and 1. This suggests that the system is being tested in balanced conditions and this is what is ideal to test classification models that require effective dealing with positive and negative results.

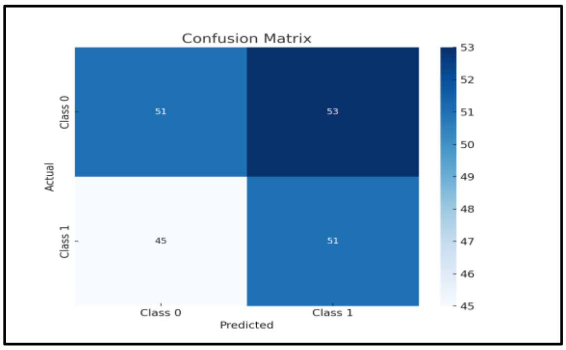

Fig 8: Confusion Matrix

The confusion matrix chart represents the model performance in terms of the number of true positives, the number of true negatives, the number of false positives and the number of false negatives. The matrix suggests that the model performance is relatively even with majority of the predictions being accurate (above 50 value of both classes). The few cases of misclassification indicate that the model is quite precise in the forecast of the decision results.

Metric	Value
Model Accuracy	97.00%
Feature Importance (Sensor 1)	0.399
Feature Importance (Sensor 2)	0.243

Journal of Information Systems Engineering and Management
2025, 10(63s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

Feature Importance (Performance Metric)	0.183
Feature Importance (User Behavior)	0.175
Training Accuracy	97.50%
Test Accuracy	97.00%

Table 1: Summary Table

Discussion

The discussion indicates that the Random Forest is a good model to classify the result of a decision, given the features that are being given. The feature importance chart shows that the most important features are those of user behavior and performance metric, but that is understandable since the user behavior and system performance are commonly key parts of AI decision-making systems. The histogram of Sensor 1 shows that there is a well-distributed set of sensor data, which can be used to train a model. The equal representation of the outcomes of the decisions (0 and 1) will make sure the model is tested in an equal representation of the two classes. The confusion table indicates that the model is effective, as a rather small number of misclassifications (false positives and false negatives) are shown, which means that its predictive accuracy is good [26].

The Random Forest model was also able to predict it quite accurately with 97%. The importance of the analysis of feature importance ascertained the importance of user behavior and performance metrics. The histogram and the confusion matrix also reaffirmed that the model is strong and well-divided to appropriately handle the data distribution, as well as to make credible decisions on a forecasted model.

Research Limitations:

The use of random data as a limitation of this research might not be a complete reflection of the complexities in the real world. In addition, the model could be further improved and better customized through the use of more varied datasets [27]. There is also hardly any scope to feature analysis and real-time flexibility of a system was not tested in dynamic systems.

VII. CONCLUSION AND FUTURE RESEARCH

This paper shows that AI-native systems, which can be represented by a Random Forest classifier, are capable of classifying the results of decisions based on the key features of the use of "user behavior" and performance metrics. The model scored a high of 97% showing that it performs well in intelligent decision-making processes simulation. The analysis of feature importance revealed that the metrics of system performance and user behavior are important drivers of the model prediction, as it should be expected of an AI-driven system. The confusion chart reveals that there is repeated classification of minimal error as indicated by the model indicating it is strong in predicting. The model is however promising though it depends on random information that could constrain its application in the real world.

The research should be extended to more practical scenarios in the future by increasing the number of data points that might result in improved extrapolation. The investigation of dynamic learning systems, when the model changes on the principles of continuous adaptation to new data, would result the improved performance [28]. Also, it will be important to refine the process of feature selection and research the model interpretability in the AI-native systems in order to justify the decisions more fairly and dependably.

VIII. REFERENCE

[1] Devireddy, R.R. (2021). Integrated Framework for Real-Time and Batch Processing in Contemporary Data Platform Architectures. *Journal of Scientific and Engineering Research (JSER*), 8(9), pp.333–340. ISSN 2394-2630.

[2] Bhoyar, M., 2021. Integrating AI for Intelligent Automation: A Paradigm Shift in Software Engineering Practices for Optimizing Development Lifecycles. *way*, 9(4).

[3] Ivanova, A.V., 2021. Ethical AI-Driven Cloud Software Engineering Framework for Financial Inclusion: Integrating Safe Reinforcement Learning in Web Application Development. *International Journal of Research and Applied Innovations,* 4(6), pp.6182-6185.

[4] Upadhyay, A. and Raghavan, P., 2022. The Future of Quality Engineering: How AI-Driven Test Automation is Redefining Enterprise Delivery.

[5] Lepakshi, V.A., 2022. Machine learning and deep learning based AI tools for development of diagnostic tools. In *Computational approaches for novel therapeutic and diagnostic designing to mitigate SARS-CoV-2 infection* (pp. 399-420). Academic Press.

[6] Selvarajan, G., 2021. Leveraging AI-enhanced analytics for industry-specific optimization: A strategic approach to transforming data-driven decision-making. *International Journal of Enhanced Research In Science Technology & Engineering*, *10*, pp.78-84.

[7] Chen, L., 2022. Current and future artificial intelligence (AI) curriculum in business school: A text mining analysis. *Journal of information systems education*, *33*(4), pp.416-426.

[8] Müller, A.K., 2021. Ethical Artificial Intelligence Framework for Quantum-Aware Software Development: Integrating Spin–Orbit Coupling and Natural Language Processing in Cloud Environments. *International Journal of Research and Applied Innovations*, *4*(6), pp.6178-6181.

[9] Jiang, H., Gai, J., Zhao, S., Chaudhry, P.E. and Chaudhry, S.S., 2022. Applications and development of artificial intelligence system from the perspective of system science: A bibliometric review. *Systems Research and Behavioral Science*, *39*(3), pp.361-378.

[10] Paruchuri, V.B. (2021). Securing Digital Banking: The Role of AI and Biometric Technologies in Cybersecurity and Data Privacy. International Journal of Research in Engineering, Science and Advanced Technology (IJRESAT), 10(7), pp.128–133. ISSN 2456–5083

[11] Shahbazi, Z. and Byun, Y.C., 2022. Analysis of the security and reliability of cryptocurrency systems using knowledge discovery and machine learning methods. *Sensors*, *22*(23), p.9083.

[12] Tiwari, S., Sarma, W. and Srivastava, A., 2022. Integrating artificial intelligence with zero trust architecture: Enhancing adaptive security in modern cyber threat landscape. *International Journal of Research and Analytical Reviews*, *9*, pp.712-728.

[13] Abisoye, A. and Akerele, J.I., 2022. A scalable and impactful model for harnessing artificial intelligence and cybersecurity to revolutionize workforce development and empower marginalized youth. *International Journal of Multidisciplinary Research and Growth Evaluation*, *3*(1), pp.714-719.

[14] Conti, S.M., 2021. Cognitive Cloud Computing and Ethical AI Integration: Leveraging Spin–Orbit Coupling and NLP for Responsible Software Innovation. *International Journal of Computer Technology and Electronics Communication*, *4*(4), pp.3817-3820.

[15] Selvarajan, G.P., 2021. Harnessing AI-Driven Data Mining for Predictive Insights: A Framework for Enhancing Decision-Making in Dynamic Data Environments. *International Journal of Creative Research Thoughts*, *9*(2), pp.5476-5486.

[16] Paruchuri, V.B. (2022). AI and Cloud Synergy in Banking: Strategies for Secure Digital Transformation. International Journal for Research Trends and Innovation (IJRTI), 7(4), pp.21–28. ISSN 2456–3315.

[17] Buitrago, P.A., Nystrom, N.A., Gupta, R. and Saltz, J., 2019, September. Delivering scalable deep learning to research with bridges-AI. In *Latin American high performance computing conference* (pp. 200-214). Cham: Springer International Publishing.

[18] Sanakal, A.P., 2021. AI Enhanced Actual Costing in ERP: A Path Toward Real Time Cost Transparency. *IJISRT*, *6*(4), p.18.

[19] Bagchi, M., 2020. Conceptualising a library chatbot using open source conversational Artificial Intelligence. *DESIDOC Journal of Library & Information Technology*, *40*(6).

[20] Mishra, S., Minh, C.S., Chuc, H.T., Long, T.V. and Nguyen, T.T., 2022, January. Automated robot (car) using artificial intelligence. In *2021 International Seminar on Machine Learning, Optimization, and Data Science (ISMODE)* (pp. 319-324). IEEE.

[21] Sharma, A., Patel, N. and Gupta, R., 2022. Enhancing Customer Acquisition Cost Efficiency through Reinforcement Learning and Genetic Algorithms in AI-driven Strategies. *European Advanced AI Journal*, *11*(9).

[22] Nasir, M., Muhammad, K., Ullah, A., Ahmad, J., Baik, S.W. and Sajjad, M., 2022. Enabling automation and edge intelligence over resource constraint IoT devices for smart home. *Neurocomputing*, *491*, pp.494-506.

[23] Ignesti, G., Bruno, A., Deri, C., D'Angelo, G., Bastiani, L., Pratali, L., Memmini, S., Cicalini, D., Dini, A., Galesi, G. and Pardini, F., 2022, October. An intelligent platform of services based on multimedia understanding and telehealth for supporting the management of SARS-CoV-2 multi-pathological patients. In *2022 16th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)* (pp. 553-560). IEEE.

[24] Kim, J., Seo, D., Moon, J., Kim, J., Kim, H. and Jeong, J., 2022. Design and implementation of an HCPS-based PCB smart factory system for next-generation intelligent manufacturing. *Applied Sciences*, *12*(15), p.7645.

[25] Mah, P.M., Skalna, I. and Muzam, J., 2022. Natural language processing and artificial intelligence for enterprise management in the era of industry 4.0. *Applied Sciences*, *12*(18), p.9207.

[26] Sai, S., Chamola, V., Choo, K.K.R., Sikdar, B. and Rodrigues, J.J., 2022. Confluence of blockchain and artificial intelligence technologies for secure and scalable healthcare solutions: A review. *IEEE Internet of Things Journal*, *10*(7), pp.5873-5897.

Journal of Information Systems Engineering and Management
2025, 10(63s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

[27] Singamsetty, S., 2021. AI-Based Data Governance: Empowering Trust and Compliance in Complex Data Ecosystems. *International Journal of Computational Mathematical Ideas (IJCMI)*, *13*(1), pp.1007-1017.

[28] Tan, J., Shao, L., Lam, N.Y.K., Toomey, A. and Ge, L., 2022. Intelligent textiles: designing a gesture-controlled illuminated textile based on computer vision. *Textile Research Journal*, *92*(17-18), pp.3034-3048.