Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Secure and Explainable AIP System Integrated with Devops Principles Using HLGSRU and SCC

Ranadheer Reddy Charabuddi
Sr. Lead SAP Opentext VIM Consultant, Avventis Inc, Northlake, Texas, USA

ARTICLE INFO

Received: 12 July 2025

Revised: 27 Aug 2025

Accepted: 07 Sept 2025

ABSTRACT

Currently, to enhance the development, deployment, monitoring, and maintenance of Automated Invoice Processing (AIP) systems, Development and Operations (DevOps) are increasingly used. Nevertheless, traditional methodologies failed to ensure the security of audit log files from unauthorized persons or intruders, leading to data leakage. Thus, this paper proposes a DevOps-based secure and explainable AIP system using Hinge Loss Gated Smish Recurrent Contrastive Unit (HLGSRU) and Shimura Curve Cryptography (SCC). Primarily, admins create invoice entries, followed by text extraction. Further, by preprocessing the invoice NER data, the Named Entity Recognition (NER) model is trained, followed by layout preservation and visual feature extraction. Concurrently, Part of Speech (PoS) tagging and contextual and syntactic pattern identification are carried out using the Hidden Skellam Markov Model (HSMM) and Nesterov Accelerated Gradient-based Financial Bidirectional Encoder Representations from Transformers (NAG-finBERT), respectively, followed by feature extraction. Now, HLGSRU performs NER classification based on the extracted features and identified patterns, followed by a deep explanation. Lastly, AIP systems are developed based on NER by securely storing audit files using SCC through a CI/CD pipeline. Therefore, the proposed framework outperforms the other conventional methodologies with higher accuracy (98.75%).

Keywords: Automated Invoice Processing (AIP), Text Extraction, Contextual and Syntactic Pattern Identification, Deep Learning (DL), Named Entity Recognition (NER), Optical Character Recognition (OCR), and Continuous Integration/Continuous Deployment (CI/CD).

Introduction

Businesses are progressively adopting structured methods to enhance the efficiency of their financial workflows, especially in handling invoice processing in today's digital environment (Kumar et al., 2023) (Hao, 2022). Invoices are important documents and serve as proof of purchase for business and tax purposes, initiating accurate invoice processing (Dragomirescu et al., 2025) (Sanchez-Obando et al., 2024). Thus, to enhance the development and maintenance of AIP systems, DevOps principles and Artificial Intelligence (AI)-based techniques are developed (Kumar et al., 2023). DevOps, which represents a cultural shift and a set of practices, ensures faster and secure handling of financial documents through CI/CD pipelines (Rohaime et al., 2022) (Ramyadevi et al., 2024). CI/CD pipelines are central to DevOps, enabling automated development, testing, and deployment of the AIP system (Zhang, 2022) (Saout et al., 2024). Yet, existing works failed to capture contextual and syntactic patterns from invoices, leading to inaccurate NER classification and decision-making (Azzam et al., 2023). Likewise, none of the conventional works secured CI/CD audit trails, which were typically stored in mutable log formats, leading to unauthorized access and tampering. Therefore, this paper is motivated to develop a DevOps principles-integrated secure and explainable AIP system using HLGSRU and SCC.

Problem Statement

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

The limitations of conventional methodologies are,

1. None of the traditional approaches prevented audit log files from unauthorized intruders or persons during AIP development, leading to potential data leakage.
2. (Ha & Horák, 2022) failed to learn contextual and syntactic patterns from the text, affecting the performance of invoice processing.
3. Due to the poor pre-processing approaches and misspelling words, (Krieger et al., 2023) provided inaccurate classification.
4. Most of the existing works failed to offer transparency in decision-making, making them unreliable for customer behavior analysis.

Objective

The proposed framework's objectives are,

1. The proposed SCC is applied to secure audit log files from unauthorized intruders or persons.
2. To learn contextual and syntactic patterns, the proposed NAG-finBERT is utilized.
3. By performing tokenization, spelling correction, and PoS tagging, the proposed framework reduces misclassification caused by poor preprocessing.
4. The proposed SHapley Townsend Additive explanation (SHTAP) is used to enhance model explainability and ensure transparent decision-making.

This paper's construction is demonstrated as: the literature survey is illustrated in Section 2, the proposed methodology is depicted in Section 3, the results and discussion are indicated in Section 4, and Section 5 demonstrates the conclusion with future recommendations.

Literature Survey

(Ha & Horák, 2022) presented a model for information extraction from invoice images. Here, for metadata extraction and rule-based block classification, this model developed the OCRMiner System for invoice component identification. Therefore, the model attained 98% accuracy. Yet, this model was unable to capture contextual and syntactic patterns due to insufficient linguistic understanding.

(Krieger et al., 2023) applied a Machine Learning (ML)-based AIP system to extract information. By using OCR, this model performed text extraction. Likewise, to extract information from layout-rich invoices, the ensemble ML approaches were utilized. The model attained ~95% accuracy. Nevertheless, due to the presence of misspelled words, this model failed to deliver reliable classification results.

(Chi et al., 2024) established a model for invoice text classification using semantic enrichment and data augmentation techniques. This model provided enriched invoice texts using class labels. Likewise, it performed text classification with ~89% accuracy. Nevertheless, due to the validation of a large number of data samples, the model's computation time was high.

(Sánchez & Cuervo-Londoño, 2024) demonstrated an information extraction framework from electricity invoices using the Bag-of-Words approach. Here, by using various ML methodologies, the information extraction was performed by extracting texts. Thus, the model attained 91.86% accuracy. Nevertheless, this model considered a small diversity of invoice documents, affecting the generalization capacity.

(Saout et al., 2022) specified a table extraction model from invoices. Here, by performing visual analysis, tables were accurately extracted for table region detection and table structure identification. Therefore, the model took a running time of 300 seconds. Nevertheless, due to misleading visual cues caused by the surrogate function, this model misclassified the empty regions as valid table regions.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Proposed Methodology for the Development of Devops-Integrated Secure and Explainable Aip System

Here, the proposed methodology for developing a DevOps-integrated secure and explainable AIP system using HLGSRU and SCC is demonstrated, and the block diagram is depicted in Figure 1.

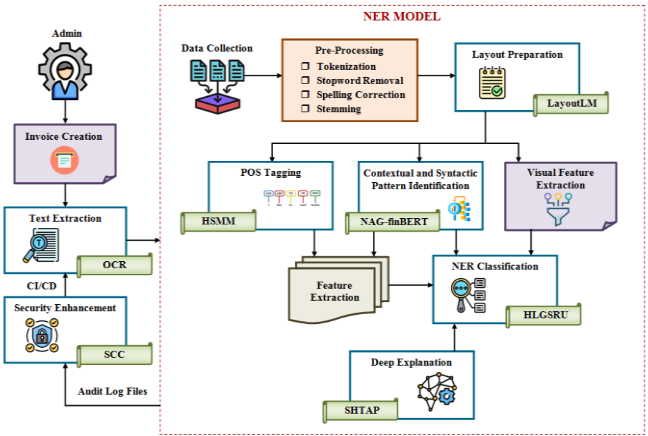


Figure 1. Block diagram for the proposed framework

Invoice Creation

Initially, invoice data $\left(\mathrm{I}_{inv}^{a}\right)$, which represents the digital document, including customer information, billing items, and billing amounts, is created by the administrators.

$$\mathrm{I}_{inv}^{a} = \left\{\mathrm{I}_{inv}^{1}, \mathrm{I}_{inv}^{2}, \mathrm{I}_{inv}^{3}, \cdots, \mathrm{I}_{inv}^{\vec{a}}\right\} \qquad ; a = 1 \rightarrow \vec{a} \tag{1}$$

Here, $\vec{a}$ specifies the maximum number of I_{inv}^{a}.

Text Extraction

Now, the OCR is applied to extract the textual information from I_{inv}^{a}. The OCR uses a scanner that reprocesses the physical form of a document into editable and digital text, thus extracting text.

Here, I_{inv}^{a} is converted into a grey-scale. Next, to correct the angular misalignments, the deskewing is applied due to improper scanning. Next, to isolate individual textual blocks, characters, or files, segmentation is performed for accurate recognition. Lastly, the OCR engine detects the characters and reconstructs the document in a structured digital format, indicating extracted text $\left(\mathrm{I}_{inv}^{\text{txt}}\right)$.

NER Model

Then, by using Invoice NER data $\left(\mathrm{I}_{ner}^{b}\right)$, the NER model is trained for accurate NER classification, helping to develop a DevOps-integrated AIP.

Data Collection

Firstly, from the publicly available sources, I_{ner}^{b} are collected.

$$\mathrm{I}_{ner}^{b} = \left\{\mathrm{I}_{ner}^{1}, \mathrm{I}_{ner}^{2}, \mathrm{I}_{ner}^{3}, \cdots, \mathrm{I}_{ner}^{\vec{b}}\right\} \qquad ; b = 1 \rightarrow \vec{b} \tag{2}$$

Here, $\vec{b}$ illustrates the maximum number of I_{ner}^{b}.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

Preprocessing

Afterward, I_{ner}^{b} often contains misspelled words, irrelevant symbols, and inconsistent formats, which affect invoice processing. Therefore, the following preprocessing steps are performed.

1. Initially, I_{ner}^{b} are decomposed into meaningful words or tokens to ensure better understanding.
2. Afterward, to retain the relevant content, the stop words or meaningless words are removed.
3. Then, to enhance textual accuracy, the misspelled words in I_{ner}^{b} are detected and corrected, ensuring better invoice processing.
4. Lastly, preprocessed data $\left(\mathrm{I}^{\mathrm{Pre}}\right)$ are obtained by performing stemming, which reduces the words to their root forms, thereby minimizing redundancy.

Layout Preservation

Later, layout preservation is conducted to maintain the structural and spatial information of $\mathrm{I}^{\mathrm{Pre}}$ using LayoutLM. The LayoutLM, which is a multi-modal pre-training method, is designed for visually rich document understanding and information extraction tasks.

Here, the bounding box coordinates $\left(B^{cor}\right)$ of tokens $\left(T\right)$ in $\mathrm{I}^{\mathrm{Pre}}$ are embedded as layout features. Now, image embedding and patch embedding are computed. Lastly, the transformer block learns spatial and linguistic context to provide layout-preserved data $\left(\mathrm{I}^{\mathrm{Lay}}\right)$.

Visual Feature Extraction

Further, the visual features $\left(F_{Vis}\right)$ that are important for accurate NER classification and effective invoice processing, including bounding box coordinates, horizontal position, vertical position, and header, are extracted from $\mathrm{I}^{\mathrm{Lay}}$.

PoS Tagging

In the meantime, PoS tagging, which indicates the process of identifying the syntactic roles of individual words in $\mathrm{I}^{\mathrm{Pre}}$, is conducted using the proposed HSMM. The Hidden Markov Model (HMM) assigns probabilities to tag sequences, enabling flexible PoS tagging. However, HMMs fail to adapt to new data due to their reliance on predefined probability distributions. Hence, the Skellam Distribution (SD) is applied, which estimates the distribution of an invoice model, especially for unseen words.

Primarily, the initial probabilities are calculated from $\mathrm{I}^{\mathrm{Pre}}$ using SD $\left(\mathrm{P}_{SD}\right)$, followed by observation sequence $\left(O_{OS}\right)$ computation using the forward algorithm.

$$\mathrm{P}_{SD}\left(\mathrm{I}^{\mathrm{Pre}}\right)=\mu_{SD}^{1}\left(D_{poi}^{1}\right)-\mu_{SD}^{2}\left(D_{poi}^{2}\right) \tag{3}$$

$$O_{OS}\left[\mathrm{P}_{SD}\right]=\sum_{h=1}^{\vec{h}} p^{for}\left(S^{hid}\right)_{h} \tag{4}$$

Here, $\left(p^{for},\vec{h}\right)$ designate the forward probabilities of hidden state $\left(S^{hid}\right)$ in HSSM and the maximum number of S^{hid}, respectively, and $\left(\mu_{SD}^{1},\mu_{SD}^{2}\right)$ illustrate the mean values of the poision variable $\left(D_{poi}^{1},D_{poi}^{2}\right)$ in SD. Now, the Viterbi algorithm calculates the maximum probabilities $\left(P^{\max}\right)$ in O_{OS}.

$$P^{\max}=\max_{S^{hid}}\left(S^{hid-1}*\mathrm{P}_{SD}\right)\times O_{OS}P_{Tran} \tag{5}$$

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Where, $\left(S^{hid-1}, P_{Tran}\right)$ exhibit the previous S^{hid} and transition probabilities of O_{OS}, respectively. Lastly, the PoS is tagged for each $\mathrm{I}^{\mathrm{Pre}}$ regarding $P^{\max}$ and O_{OS} and is denoted as P_{Pos}.

Contextual and Syntactic Pattern Identification

Meanwhile, the contextual and syntactic patterns are identified from $\mathrm{I}^{\mathrm{Pre}}$ using NAG-finBERT for accurate NER classification. Traditional financial Bidirectional Encoder Representations from Transformer (finBERT) is a BERT-based model fine-tuned for financial text, enabling better understanding of financial terminology. Yet, finBERT leads to suboptimal results due to poor padding schemes. Therefore, the Nesterov Accelerated Gradient (NAG) rule, which dynamically adjusts the learning trajectory, is applied to set the padding parameters.

Chiefly, based on $\mathrm{I}^{\mathrm{Pre}}$, the padding parameters of NAG-finBERT are initialized using NAG.

$$\mathrm{P}_{ini} = m^{coe} m^{vel} + r_{lea} \Delta^{gra} \left(\omega_{par} - m^{coe} m^{vel}\right) \tag{6}$$

Here, $\left(\mathrm{P}_{ini}, m^{coe}, m^{vel}, r_{lea}, \Delta^{gra}, \omega_{par}\right)$ demonstrate initialized parameters, momentum coefficient, momentum velocity, learning rate, gradient loss function, and weight parameter of NAG-finBERT, correspondingly. Now, $\mathrm{I}^{\mathrm{Pre}}$ are embedded $\left(\widetilde{\mathrm{I}}^{\mathrm{Pre}}\right)$ into a semantic vector space based on P_{ini}, followed by contextual representation learning using multiple transformer blocks $\left(\mathrm{T}_{mul}\right)$.

$$\mathrm{T}_{mul} = \sum \omega^{\widetilde{\mathrm{I}}^{\mathrm{Pre}}} \cdot v_{\widetilde{\mathrm{I}}^{\mathrm{Pre}}} \tag{7}$$

Where, $\left(\omega^{\widetilde{\mathrm{I}}^{\mathrm{Pre}}}, v_{\widetilde{\mathrm{I}}^{\mathrm{Pre}}}\right)$ reveal the weight value of T_{mul} and the vector values computed from $\widetilde{\mathrm{I}}^{\mathrm{Pre}}$, correspondingly. Then, by using T_{mul}, the feed-forward layer updates $\widetilde{\mathrm{I}}^{\mathrm{Pre}}$. Finally, fine-tuning provides the identified contextual and syntactic patterns $\left(P_{CSP}\right)$.

$$\widetilde{\mathrm{I}}_{U}^{\mathrm{Pre}} = \sum \mathrm{T}_{mul}\left(\widetilde{\mathrm{I}}^{\mathrm{Pre}}\right) \tag{8}$$

$$P_{CSP} = \widetilde{\mathrm{I}}_{U}^{\mathrm{Pre}} - \ell^{r} \cdot \Delta_{gl}\left(\widetilde{\mathrm{I}}_{U}^{\mathrm{Pre}}\right) \tag{9}$$

Here, $\left(\widetilde{\mathrm{I}}_{U}^{\mathrm{Pre}}, \ell^{r}, \Delta_{gl}\right)$ represent the updated $\widetilde{\mathrm{I}}^{\mathrm{Pre}}$, learning rate, and gradient of loss, correspondingly.

Feature Extraction

Now, for the estimation of NER classification and invoice processing, the features from P_{CSP} and P_{Pos} are extracted. Here, PoS Tag of Each Word, PoS Tag Sequency, and Sentence type are extracted from P_{Pos}. Likewise, features like token text, token embedding, word length, and font size are extracted from P_{CSP}.

$$F_{fea}^{i} = \left\{F_{fea}^{1}, F_{fea}^{2}, F_{fea}^{3}, \cdots, F_{fea}^{\vec{i}}\right\} \qquad ; i = 1 \to \vec{i} \tag{10}$$

Here, $\vec{i}$ designates the maximum number of extracted features $\left(F_{fea}^{i}\right)$ from $\left(P_{CSP}, P_{Pos}\right)$.

NER Classification

Later, by using HLGSRU, the classification model is trained based on $\left(F_{fea}, F_{Vis}, P_{CSP}\right)$ for the NER accurate classification. Gated Recurrent Unit (GRU) uses less memory and can process large input quickly. Yet, GRU has overfitting and low-learning efficiency issues, affecting the performance in NER classification. Hence, the Hinge Loss Regularization (HLR) is applied to learn the scarcity pattern

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ Research Article

dynamically and perform feature-wise regularization, enhancing generalization by eliminating the overfitting issue. Also, the Smish activation function is used to handle long dependencies in financial sequences and avoid dead neurons, improving learning efficiency. In Figure 2, the classifier diagram for HLGSRU is depicted.

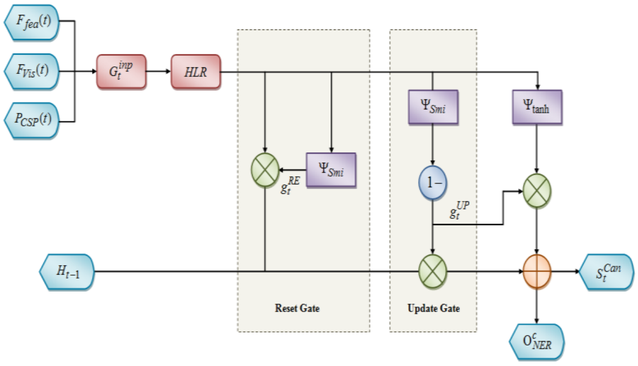


Figure 2. Classifier diagram for HLGSRU

1. Firstly, the previous hidden state (H_{t-1}) is represented at the time step (t) for each $(F_{fea}, F_{Vis}, P_{CSP})$.
2. Afterward, to perform feature-wise regularization on $(F_{fea}, F_{Vis}, P_{CSP})$, the HLR $(\Re^{HLR})$ is applied.

$$G_t^{inp} = \begin{cases} F_{fea}(t) \\ F_{Vis}(t) \\ P_{CSP}(t) \end{cases} \tag{11}$$

$$\Re^{HLR}(G_t^{inp}) = \min \frac{1}{2}\left\|\omega_{\Re^{HLR}}\right\|^2 + p_{reg}\Sigma\max\left(0, 1 - G_t^{inp}\left((\omega_{\Re^{HLR}})^{tra} G_t^{inp} + \upsilon_{\Re^{HLR}}\right)\right) \tag{12}$$

Here, $(G_t^{inp}, p_{reg}, tra, \omega_{\Re^{HLR}}, \upsilon_{\Re^{HLR}})$ exemplify the input of HLGSRU, regularization parameter, transpose function, weight, and bias value of $\Re^{HLR}$, respectively.

3. Then, to control past information flow, the reset gate (g_t^{RE}) and update gate (g_t^{UP}) are computed with the Smish activation function (Ψ_{Smi}).

$$g_t^{RE} = \left[\omega^{g_t^{RE}}\left(H_{t-1} \cdot \Re^{HLR}\right) + \upsilon^{g_t^{RE}}\right] \times \Psi_{Smi} \tag{13}$$

$$g_t^{UP} = \left\{\upsilon^{g_t^{UP}} + 1 - \left[\omega^{g_t^{UP}}\left(H_{t-1} \cdot \Re^{HLR}\right)\right]\right\} \times \Psi_{Smi} \tag{14}$$

$$\Psi_{Smi}(\Re^{HLR}) = s_{coe}\Re^{HLR} \cdot \tanh\left[\ln\left(1 + \frac{1}{1 + e^{-s_{par}\Re^{HLR}}}\right)\right] \tag{15}$$

Where, $(\omega^{g_t^{RE}}, \omega^{g_t^{UP}})$ designate the weight values of (g_t^{RE}, g_t^{UP}), respectively, $(\upsilon^{g_t^{RE}}, \upsilon^{g_t^{UP}})$ reveal the bias values of (g_t^{RE}, g_t^{UP}), correspondingly, and (s_{coe}, s_{par}) illustrate the scaling coefficient and scaling parameter, respectively.

4. Afterward, by using the tanh activation function $(\Psi_{\tanh})$, the hidden (S_t^H) and candidate hidden states' (S_t^{Can}) outputs are calculated, attaining a classified outcome (O_{NER}^c).

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

$$S_t^{Can} = \Psi_{\tanh}\left(\omega^{S_t^H}\left[g_t^{RE} \times H_{t-1}; \Re^{HLR}\right] + \upsilon^{S_t^H}\right) \tag{16}$$

$$S_t^H = H_{t-1} - g_t^{UP} * H_{t-1} + g_t^{UP} S_t^{Can} \tag{17}$$

$$\Psi_{\tanh}\left(\Re^{HLR}\right) = \frac{\exp\left(\Re^{HLR}\right) - \exp\left(-\Re^{HLR}\right)}{\exp\left(\Re^{HLR}\right) + \exp\left(-\Re^{HLR}\right)} \tag{18}$$

$$O_{NER}^c = \left\{O_{NER}^1, O_{NER}^2, O_{NER}^3, \cdots, O_{NER}^{\vec{c}}\right\} \qquad ; c = 1 \to \vec{c} \tag{19}$$

Here, $\left(\omega^{S_t^H}, \upsilon^{S_t^H}\right)$ represent the weight and bias value of S_t^{Can}, correspondingly, and $\vec{c}$ illustrates the number of classified NER (pay to, invoice number, total amount, and so on).

The pseudo-code for HLGSRU is depicted as follows,

Pseudo-code for HLGSRU

Input: Features $\left(F_{fea}, F_{Vis}, P_{CSP}\right)$

Output: Classified Outcome O_{NER}^c

Begin

Initialize $G_t^{inp}, p_{reg}, g_t^{RE}, g_t^{UP}$, iteration (I), and maximum iteration (I_{max}).

Set $(I = 1)$

While $(I \le I_{max})$

For each G_t^{inp} do

Perform Feature-wise regularization

$$\Re^{HLR}\left(G_t^{inp}\right) = \min \frac{1}{2}\left\|\omega_{\Re^{HLR}}\right\|^2 + p_{reg}\sum \max\left(0, 1 - G_t^{inp}\left(\left(\omega_{\Re^{HLR}}\right)^{tra} G_t^{inp} + \upsilon_{\Re^{HLR}}\right)\right)$$

Use $\left(\Psi_{Smi}\right)$

$$\Psi_{Smi}\left(\Re^{HLR}\right) = s_{coe}\Re^{HLR} \cdot \tanh\left[\ln\left(1 + \frac{1}{1 + e^{-s_{par}\Re^{HLR}}}\right)\right]$$

Compute $\left(g_t^{RE}, g_t^{UP}\right)$

$$g_t^{RE} = \left[\omega^{g_t^{RE}}\left(H_{t-1} \cdot \Re^{HLR}\right) + \upsilon^{g_t^{RE}}\right] \times \Psi_{Smi}$$

$$g_t^{UP} = \left\{\upsilon^{g_t^{UP}} + 1 - \left[\omega^{g_t^{UP}}\left(H_{t-1} \cdot \Re^{HLR}\right)\right]\right\} \times \Psi_{Smi}$$

Calculate $\left(S_t^H, S_t^{Can}\right)$

$$S_t^{Can} = \Psi_{\tanh}\left(\omega^{S_t^H}\left[g_t^{RE} \times H_{t-1}; \Re^{HLR}\right] + \upsilon^{S_t^H}\right)$$

$$S_t^H = H_{t-1} - g_t^{UP} * H_{t-1} + g_t^{UP} S_t^{Can}$$

End For

End While

Return: $O_{NER}^c = \left\{O_{NER}^1, O_{NER}^2, O_{NER}^3, \cdots, O_{NER}^{\vec{c}}\right\}$

End

Therefore, the proposed HLGSRU accurately classifies the NER.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

Explanation

Now, SHTAP is used to provide a deep explanation about O_{NER}^{c}, providing transparency and interpretable insights into model decisions. SHapley Additive explanation (SHAP) ensures consistency across models and enhances trust and transparency. Nevertheless, SHAP provides misleading results due to the use of an improper weight assignment function. Hence, the Townsend Function (TF) is used to assign feature weights based on feature relevance and contextual impact.

Chiefly, the SHAP values or features (κ^{fea}) are computed from O_{NER}^{c}, followed by weight assignment using the TF (ξ_{TF}).

$$\kappa^{fea} = \sum \frac{1}{\vec{d}!}\left(\frac{|b^{sss}|!(\vec{d} - |b^{sss}| - 1)!}{\left[O_{NER}^{c}(b^{sss}) - O_{NER}^{c}(b^{sss})_{\vec{d}} \right]^{-1}} \right) \qquad ; d = 1 \rightarrow \vec{d} \tag{20}$$

$$\xi_{TF}(\kappa_{u}^{fea}, \kappa_{v}^{fea}) = -(\cos([\kappa_{u}^{fea} - 0.1]\kappa_{v}^{fea}))^{2} - \kappa_{u}^{fea}\sin(3\kappa_{u}^{fea} + \kappa_{y}^{fea}) \tag{21}$$

Here, $(b^{sss}, \vec{d})$ exemplify the simplified binary κ^{fea} and maximum number of κ^{fea}, correspondingly, and $(\kappa_{u}^{fea}, \kappa_{v}^{fea})$ portray u_{th} and v_{th} SHAP feature, respectively. Finally, based on ξ_{TF} and base value (κ^{base}), the explanation model $(M^{\exp})$ is defined, providing an efficient explanation about O_{NER}^{c}.

$$M^{\exp} = \kappa^{base} + \sum_{d=1}^{\vec{d}} [\xi_{TF}]_{d} (b^{sss})_{d} \tag{22}$$

Therefore, the proposed SHTAP offers interpretable model explanations.

Later, AIP system is developed through CI/CD pipelines based on O_{NER}^{c} and $M^{\exp}$. During this process, the audit log files (X^{aud}), which are crucial records that track every action and event in CI/CD pipelines, are created.

Security Enhancement

Lastly, the proposed SCC is used to store X^{aud} securely from unauthorized persons or intruders through CI/CD pipelines to ensure confidentiality and integrity of X^{aud}. Elliptic Curve Cryptography (ECC) offers a high level of security with relatively smaller key sizes. Yet, ECC leads to high computational cost due to the negative points in the elliptic curve. Hence, the Shimura Curve (SC), which provides enhanced cryptographic operations by eliminating the computational inefficiencies, is used.

1. Primarily, by using the SC (ξ^{SC}), the cryptographic operations are performed, followed by private $(K^{\Pr})$ and public key (K^{Pu}) generation with respect to X^{aud}.

$$\xi^{SC} = x^{2} + y^{2} + 3z^{2} \tag{23}$$

$$K^{\Pr} = K^{\mathrm{Pu}} \cdot P_{ini}(\xi^{SC}) \tag{24}$$

Where, P_{ini} and (x, y, z) establish the initial point and projective coordinates on ξ^{SC}, respectively.

2. Next, the ciphertexts (C^{t1}, C^{t2}) are generated based on $(K^{\Pr}, K^{\mathrm{Pu}})$ to enhance data security, attaining encrypted X^{aud} (X_{enc}^{aud}).

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

$$C^{t1} = P^{ran} \cdot P_{ini} \tag{25}$$

$$C^{t2} = \left(\mathrm{X}^{aud} + P^{ran}\right) * \left(\mathrm{X}^{aud} + K^{\mathrm{Pu}}\right) \tag{26}$$

Here, P^{ran} portrays the random point in ξ^{SC}.

3. Lastly, the original information of X^{aud}_{enc} is obtained by performing decryption.

$$\mathrm{X}^{aud} = C^{t2} - K^{\mathrm{Pr}} \times C^{t1} \tag{27}$$

The proposed SCC's pseudo-code is depicted as follows,

Pseudo-code for SCC

Input: Audit Files X^{aud}
Output: Encrypted data X^{aud}_{enc}

Begin
    Initialize $K^{\mathrm{Pr}}, K^{\mathrm{Pu}}, P^{ran}, P_{ini}$, iteration (I), and maximum iteration $(\mathrm{I}_{\max})$.
    Set $(\mathrm{I} = 1)$
    While $(\mathrm{I} \le \mathrm{I}_{\max})$
        For each X^{aud} do
            Use SC
                $\xi^{SC} = x^2 + y^2 + 3z^2$
            Find $\left(K^{\mathrm{Pr}}, K^{\mathrm{Pu}}\right)$
                $K^{\mathrm{Pr}} = K^{\mathrm{Pu}} \cdot P_{ini}\left(\xi^{SC}\right)$
            Perform Encryption
            Generate $\left(C^{t1}, C^{t2}\right)$
                $C^{t1} = P^{ran} \cdot P_{ini}$
                $C^{t2} = \left(\mathrm{X}^{aud} + P^{ran}\right) * \left(\mathrm{X}^{aud} + K^{\mathrm{Pu}}\right)$
            Apply Decryption
                $\mathrm{X}^{aud} = C^{t2} - K^{\mathrm{Pr}} \times C^{t1}$
        End For
    End While
    Return: X^{aud}_{enc}
End

Therefore, in DevOps-integrated AIP development, the proposed SCC ensures end-to-end encryption with confidentiality and integrity.

Result And Discussion

Here, the proposed framework is evaluated, and the implementation is conducted in the PYTHON platform.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

Dataset Description

The proposed framework uses the Invoice NER dataset, and the source link is depicted in the reference section. This dataset, which originally contains 67 invoices of data, is particularly designed for NER and information extraction. After augmentation, this dataset expands to 1005 instances. Among them, 80% (804) data are wielded for training, and the remaining 20% (201) data are used for testing.

Performance Validation

Here, the proposed framework is compared with several conventional methodologies.

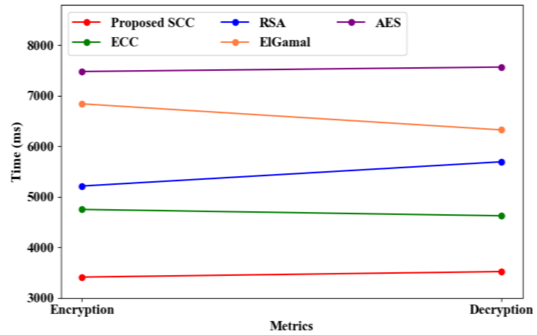
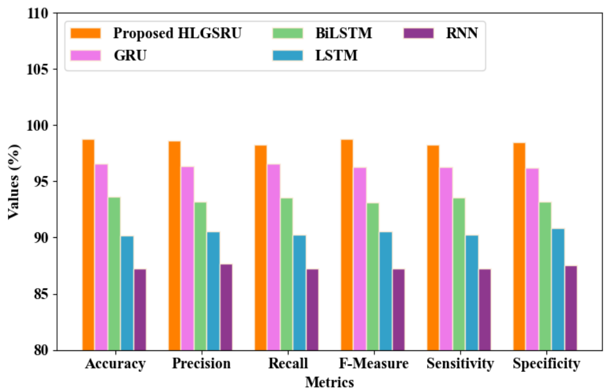


Figure 3. Time Analysis

Table 1. Efficiency analysis for SSC

Techniques	Encryption Time (ms)	Decryption Time (ms)	Security Level (%)
Proposed SCC	3412	3521	98.2345
ECC	4751	4625	95.7481
RSA	5214	5694	92.1545
ElGamal	6841	6325	89.7846
AES	7481	7569	87.3256

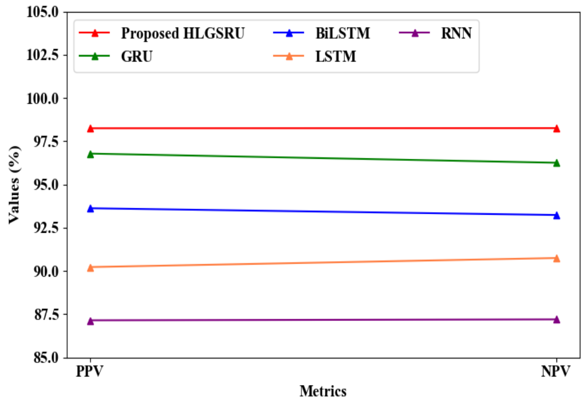
The SCC's performance in data security enhancement is depicted in Figure 3 and Table 1. The SSC significantly outperforms traditional methodologies like ECC, Rivest, Shamir, Adleman (RSA), ElGamal, along with Advanced Encryption Standard (AES) centered on encryption time (3412ms), decryption time (3521ms), and security level (98.2345%). The SCC takes a minimum encryption time and decryption time while achieving the highest security level during AIP system development using Shimura curves, which reduce the computation complexity while enhancing data confidentiality. Yet, the traditional methodologies require longer computational times and provide comparatively lower security levels due to their reliance on complex arithmetic operations, reducing data security.



(a)

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article



(b)

Figure 4(a) and (b). Performance evaluation for HLGSRU

Table 2. Efficacy Evaluation for HLGSRU

Techniques	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Sensitivity (%)	Specificity (%)	PPV (%)	NPV (%)
Proposed HLGSRU	98.75	98.66	98.25	98.75	98.24	98.46	98.25	98.25
GRU	96.59	96.33	96.58	96.25	96.25	96.21	96.78	96.25
BiLSTM	93.66	93.21	93.52	93.12	93.55	93.21	93.62	93.24
LSTM	90.21	90.58	90.24	90.58	90.26	90.85	90.23	90.75
RNN	87.25	87.66	87.21	87.24	87.21	87.56	87.15	87.20

HLGSRU achieves higher accuracy (98.75%), precision (98.66%), recall (98.25%), f-measure (98.75%), sensitivity (98.24%), specificity (98.46%), Positive Predictive Value (PPV) (98.25%), along with Negative Predictive Value (NPV) (98.25%) in NER classification, as given in figure 4(a) and (b) and Table 2. The HLGSRU model effectively enhances generalization, reduces overfitting, and improves learning efficiency in NER classification by using HLR and Simish activation functions. Nevertheless, the traditional GRU, Bidirectional Long-Short Term Memory (LSTM), LSTM, and Recurrent Neural Network (RNN) fail to achieve better performance due to limited handling of long-term dependencies, resulting in reduced classification accuracy than the proposed framework.

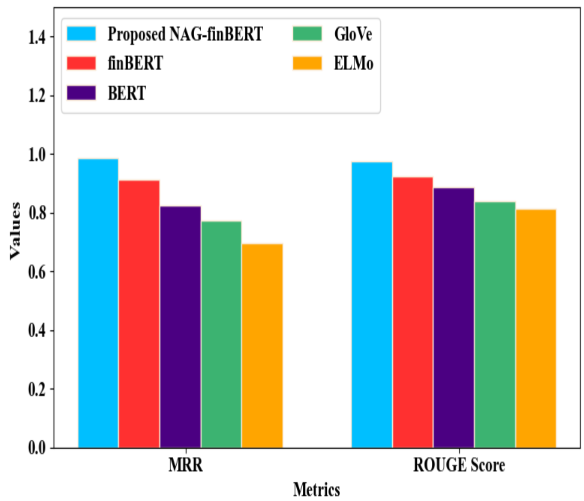


Figure 5. MRR and ROUGE analysis

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

Table 3. Performance Validation for NAG-finBERT

Techniques	MRR	ROUGE Score	EM Score
Proposed NAG-finBERT	0.9854	0.9758	5.9865
finBERT	0.9124	0.9254	4.2584
BERT	0.8236	0.8874	3.6589
GloVe	0.7745	0.8412	2.8475
ELMo	0.6958	0.8124	1.8457

In Figure 5 and Table 3, the NAG-finBERT's performance in contextual and syntactic pattern analysis is evaluated. Here, the traditional methodologies like finBERT, BERT, Global Vector for word representation (GloVe), and Embeddings from Language Models (ELMo) achieve significantly lower scores in Mean Reciprocal Rank (MRR), Recall-Oriented Understudy for Gisting Evaluation (ROUGE), and Exact Match (EM) score. While, the NAG-finBERT improves the model's ability by using the NAG to set the padding parameters. Therefore, the NAG-finBERT outperforms the traditional methodologies by attaining higher MRR (0.9854), ROUGE score (0.9758), and EM score (5.9865), improving NER classification.

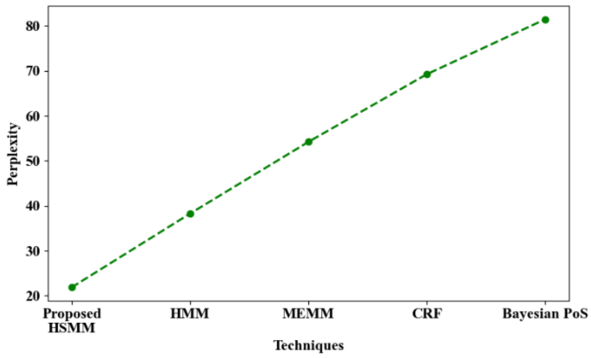


Figure 6. Perplexity Evaluation

The perplexity evaluation of HSMM and traditional methodologies like HMM, Maximum Entropy Markov Model (MEMM), Conditional Random Field (CRF), and Bayesian PoS for PoS tagging are depicted in Figure 6. Perplexity reflects how well the model tags in a sequence, where lower values indicate better generalization performance. The proposed HSMM attains lower perplexity (21.8475) due to the SD function to learn sparse and variable data patterns from invoice data, resulting in superior performance in handling complex linguistic structures. Yet, traditional HMM, MEMM, CRF, and Bayesian PoS attain higher perplexity (38.2589, 54.2158, 69.2584, and 81.4723, respectively) due to limited adaptability to irregular linguistic patterns. Therefore, the proposed HSMM outperforms the traditional methodologies.

Comprehensive Analysis

Here, the proposed framework is compared with traditional works.

Table 4. Comparative Analysis

Author's Name	Methodologies	F-measure (%)	Precision (%)
Proposed Framework	HLGSRU	98.75	98.66
(Amari et al., 2024)	RetinaNet	56	68
(Baviskar et al., 2017)	BiLSTM-Conditional Random Field	93	98

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

	(BiLSTM-CRF)		
(Loukil et al., 2024)	OCR	92	95
(Krieger et al., 2021)	GRU	89.63	93.91
(Arslan, 2022)	You Only-Look Once Version-5 (YOLOv5)	94.37	95.65

The proposed HGSRU outperforms all traditional approaches in NER classification for AIP system development with higher f-measure (98.75%) and precision (98.66%), as given in Table 4. (Amari et al., 2024) show comparatively very low performance with 56% f-measure and 68% precision due to the lack of contextual dependencies although traditional approaches like RetinaNet, BiLSTM-CRF, and YOLOv5 offer better performance in information extraction and table extraction. Similarly, all the other prevailing techniques show limited performance, highlighting HGSRU's efficacy in NER classification.

Conclusion

Here, by using HLGSRU and SCC, a security and explainability-enhanced AIP system was developed by integrating DevOps principles. Here, HSMM effectively performed PoS tagging with 21.8475 perplexities. Besides, NAG-finBERT attained 0.9758 ROUGE for contextual and syntactic patterns analysis. Further, by using HLGSRU, named entities were accurately classified with 98.74% accuracy, and the model interpretability was enhanced using SHTAP. Lastly, by using SCC, CI/CD audit files' security was enhanced with a 98.23% security level. Therefore, the proposed framework outperformed the other traditional methodologies.

Future Scope: The proposed framework effectively developed a secure DevOps-integrated AIP system. Thus, the work will be extended in the future using multimodal transformer models and multilingual Natural Language Processing (NLP) pipelines for better invoice understanding across diverse formats and languages.

Reference

Dataset: https://www.kaggle.com/datasets/nikitpatel/invoice-ner-dataset

[1] Amari, A., Makni, M., Fnaich, W., Lahmar, A., Koubaa, F., Charrad, O., Zormati, M. A., & Douss, R. Y. (2024). An Efficient Deep Learning-Based Approach to Automating Invoice Document Validation. Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA, 1–8. https://doi.org/10.1109/AICCSA63423.2024.10912544

[2] Arslan, H. (2022). End to End Invoice Processing Application Based on Key Fields Extraction. IEEE Access, 10, 78398–78413. https://doi.org/10.1109/ACCESS.2022.3192828

[3] Azzam, F., Jaber, M., Saies, A., Kirresh, T., Awadallah, R., Karakra, A., Barghouthi, H., & Amarneh, S. (2023). The Use of Blockchain Technology and OCR in E-Government for Document Management: Inbound Invoice Management as an Example. Applied Sciences (Switzerland), 13(14), 1–15. https://doi.org/10.3390/app13148463

[4] Baviskar, D., Ahirrao, S., & Kotecha, K. (2021). Multi-Layout Invoice Document Dataset (MIDD): A Dataset for Named Entity Recognition. Data, 6(7), 1–10. https://doi.org/10.3390/data6070078

[5] Baviskar, Di., Ahirrao, S., & Kotecha, K. (2017). Multi-Layout Unstructured Invoice Documents Dataset: A Dataset for Template-Free Invoice Processing and Its Evaluation Using AI Approaches. IEEE Access, 9, 1–24. https://doi.org/10.1109/ACCESS.2021.3096739

[6] Chi, W. W., Tang, T. Y., Salleh, N. M., Mukred, M., Alsalman, H., & Zohaib, M. (2024). Data Augmentation With Semantic Enrichment for Deep Learning Invoice Text Classification. IEEE Access, 12, 57326–57344. https://doi.org/10.1109/ACCESS.2024.3387860

[7] Dragomirescu, O. A., Crăciun, P. C., & Bologa, A. R. (2025). Enhancing Invoice Processing Automation Through the Integration of DevOps Methodologies and Machine Learning. Systems,

13(2), 1–23. https://doi.org/10.3390/systems13020087
[8] Ha, H. T., & Horák, A. (2022). Information extraction from scanned invoice images using text analysis and layout features. Signal Processing: Image Communication, 1–16. https://doi.org/10.1016/j.image.2021.116601
[9] Krieger, F., Drews, P., & Funk, B. (2023). Automated invoice processing: Machine learning-based information extraction for long tail suppliers. Intelligent Systems with Applications, 20, 1–14. https://doi.org/10.1016/j.iswa.2023.200285
[10] Krieger, F., Drews, P., Funk, B., & Wobbe, T. (2021). Information Extraction from Invoices: A Graph Neural Network Approach for Datasets with High Layout Variety. Lecture Notes in Information Systems and Organisation, 1–18. https://doi.org/10.1007/978-3-030-86797-3_1
[11] Kumar, N., Katoch, S., Tripathi, S., Singh, B. P., Sajal, S., & Yadav, A. L. (2023). AI Enabled Invoice Management Application. Proceedings of the 8th International Conference on Communication and Electronics Systems, ICCES 2023, 1–6. https://doi.org/10.1109/ICCES57224.2023.10192802
[12] Kumar, R., Malholtra, R. K., Singh,
[13] R., Kathuria, S., Balyan, R., & Pal, P. (2023). Artificial Intelligence Role in Electronic Invoice Under Goods and Services Tax. In 2023 International
[14] Conference on Computational Intelligence, Communication Technology and Networking CICTN, 140-143. https://ieeexplore.ieee.org/abstract/document/10140870/
[15] Loukil, F., Cadereau, S., Verjus, H., Galfre, M., Salamatian, K., Telisson, D., Kembellec, Q., & Le Van, O. (2024). LLM-centric pipeline for information extraction from invoices. 2024 2nd International Conference on Foundation and Large Language Models, FLLM 2024, 1–7. https://doi.org/10.1109/FLLM63129.2024.10852504
[16] Ramyadevi, R., Anbuselvan, A., Julian, A., Selvi, S., & Murali Dhar, M. S. (2024). Invoice Processing Automation Using UI Path Studio. Proceedings - International Conference on Computing, Power, and Communication Technologies, IC2PCT 2024, 486–489. https://doi.org/10.1109/IC2PCT60090.2024.10486692
[17] Rohaime, N. A., Abdul Razak, N. I., Thamrin, N. M., & Shyan, C. W. (2022). Integrated Invoicing Solution: A Robotic Process Automation with AI and OCR Approach. 2022 IEEE 20th Student Conference on Research and Development (SCOReD), November 8-9, 2022, Bangi, Malaysia, 1–5. https://doi.org/10.1109/SCOReD57082.2022.9973841
[18] Sánchez, J., & Cuervo-Londoño, G. A. (2024). A Bag-of-Words Approach for Information Extraction from Electricity Invoices. AI, 5, 1837–1857. www.preprints.org
[19] Sanchez-Obando, J. W., Duque-Méndez, N. D., & Bedoya Herrera, O. M. (2024). Construction of the Invoicing Process through Process Mining and Business Intelligence in the Colombian Pharmaceutical Sector. Computers, 13(10), 1–25. https://doi.org/10.3390/computers13100245
[20] Saout, T., Lardeux, F., & Saubion, F. (2022). A Two-stage Approach for Tables Extraction in Invoices. Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI, 1–14. https://doi.org/10.1109/ICTAI59109.2023.00010
[21] Saout, T., Lardeux, F., & Saubion, F. (2024). An Overview of Data Extraction from Invoices. IEEE Access, 12, 19872–19886. https://doi.org/10.1109/ACCESS.2024.3360528
[22] Zhang, H. (2022). A deep learning model for ERP
[23] enterprise financial management system. Advances in Multimedia, 2022(1), 1-11. https://doi.org/10.1155/2022/5783139