Journal of Soft Computing and Computer Applications

Manuscript 1024

Intelligent Extensible Markup Language Encryption Using Type-2 Fuzzy Logic

Faiez Musa Lahmood AlRufaye

Seham Ahmed Hashem

JOURNAL OF SOFT COMPUTING AND COMPUTER APPLICATIONS 2025;2:1024

ORIGINAL STUDY

Intelligent Extensible Markup Language Encryption Using Type-2 Fuzzy Logic

Faiez Musa Lahmood AlRufaye *, **Seham Ahmed Hashem**

Middle Technical University, Technical Instructors Training Institute, Institute St., Al-Zafaraniyah District, 10074, Baghdad, Iraq

ABSTRACT

Financial and commercial institutions increasingly rely on Extensible Markup Language (XML) files as a standard means of exchanging data. However, this extensive use has created serious security challenges due to the fact that these files contain sensitive information such as bank card numbers and expiration dates. Relying on traditional full file encryption methods achieves a high degree of security, but it causes problems related to the large file sizes that consume memory and the long encryption and decryption times, which reduces the efficiency of systems when dealing with a large number of daily transactions. Methods based on Type-1 Fuzzy Logic System (T1FLS) have failed to address the ambiguity and uncertainty inherent in data, which can lead to the misclassification of some sensitive elements. Based on this gap, this research proposes an intelligent model based on Type-2 Fuzzy Logic System (T2FLS) to classify XML file components and determine their security importance more accurately. Partial encryption can then be applied only to the parts classified as highly sensitive. The proposed model was evaluated through a series of experiments on financial files in XML formats. The results showed that it reduced encryption time by 49% compared to full encryption, reduced the size of the encrypted file by approximately 60%, and reduced memory consumption by 32%. The model also demonstrated strong resistance to common security attacks. This research confirms that combining type-2 fuzzy logic with standard cryptographic techniques achieves a balance between efficiency and security, making it suitable for application in banking and e-commerce systems.

Keywords: Extensible markup language, Type-2 fuzzy logic, Encryption, Intelligent

1. Introduction

Extensible Markup Language (XML) is widely adopted for representing and exchanging structured data in service-oriented and financial information systems due to its platform-independent syntax and support for schema-based verification and consistency [1]. In online banking, payment gateways, and web-based financial workflows, sensitive

Received 3 September 2025; revised 22 December 2025; accepted 22 December 2025.
Available online 26 December 2025

* Corresponding author.
E-mail addresses: faiez.alrufaye@mtu.edu.iq (F. M. L. AlRufaye), seham.ahmed@mtu.edu.iq (S. A. Hashem).

https://doi.org/10.70403/3008-1084.1024

information such as account IDs, card numbers, and authorization codes is routinely embedded in XML documents and transmitted across open networks, making strong message-level security essential [2]. The World Wide Web Consortium (W3C) XML cryptography specification defines standard mechanisms for encrypting entire XML documents or specific elements using symmetric and asymmetric encryption algorithms and has become the cornerstone of XML-based confidentiality services [3].

One straightforward way to protect financial XML messages is to implement full-stack encryption, where the entire document is encrypted using strong encryption such as Advanced Encryption Standard (AES) in a manner compliant with W3C standards [1]. Although this strategy provides a uniform level of security, it also introduces latency, significant memory consumption, and additional bandwidth costs, and prevents intermediaries from accessing non-sensitive fields necessary for routing, logging, or compliance verification. To mitigate these drawbacks, recent research has explored selective encryption or element-based encryption, where only XML elements containing sensitive information are encrypted. This approach aims to maintain interoperability and reduce additional costs while preserving strong protection for critical fields [2].

Recent studies in secure data processing and the Internet of Things (IoT) show that fuzzy-guided encryption and access control systems can improve the balance between security and computing efficiency compared to static, rule-based strategies [3, 4]. In particular, comparative analyses of type-2 fuzzy logic-based encryption models indicate enhanced data security and resilience when protecting resource-constrained IoT devices [5]. However, most current XML-oriented models either rely on type-1 fuzzy logic or use precise classification rules, which address tag sensitivity uncertainties in a limited way and may struggle to handle fuzzy, ambiguous, or sophisticated transaction patterns.

Type-1 Fuzzy Logic Systems (T1FLSs) have long been used to describe linguistic concepts such as “large quantity” or “frequent use” with graded membership values, but they assume a single, precise membership degree for each input, limiting their ability to capture higher degrees of uncertainty. Type-2 Fuzzy Logic Systems (T2FLSs) generalize this idea by introducing uncertainty into the membership functions themselves, which are typically represented by type-2 fuzzy sets with an interval and an uncertainty signature [5]. Recent research has shown that well-designed T2FLSs can better handle uncertainty, noise, and instability in financial and control applications compared to their type-1 counterparts. For example, Jankova and Rakowska demonstrate that different designs of type-2 membership functions significantly affect the forecasting quality of international financial markets, highlighting the role of the Footprint of Uncertainty (FoU) in capturing fuzzy expert knowledge [4]. Raj and Yang analytically examine Type-2 dual-term fuzzy controllers, demonstrating how changing the FoU impacts the balance between performance and robustness, thus highlighting the importance of Type-2 structured design [6].

Inspired by these developments, this research proposes an intelligent XML cipher model that utilizes a T2FLS classifier to achieve standards-compliant partial encryption for financial transaction messages. Intelligent classification mechanisms have been proposed to determine which XML elements should be encrypted and with what strength. Fuzzy logic and machine learning models can associate the semantic and contextual features of each tag (e.g., whether it contains credentials, financial amounts, or customer IDs) with a graded “sensitive” or “important” score that drives the encryption policy. In the proposed model, each XML element is described by a set of security-related linguistic variables (such as confidentiality effect, organizational significance, and access frequency), and a type-2 fuzzy inference system associates these inputs with a continuous sensitivity score. The uncertainty effect in the classifier’s membership functions allows for explicit encoding of ambiguity at the boundary between, for example, “medium sensitivity” and “high

sensitivity" fields, something difficult to capture using type-1 fuzzy sets. Elements whose sensitivity scores exceed a configurable threshold are encrypted using AES according to the W3C XML cipher standard, while less sensitive elements remain in plain text to maintain interoperability and reduce computational costs. This architecture directly addresses the limitations of type-1 fuzzy and clear classifiers by providing an uncertainty-aware decision surface that more smoothly adapts to the turbulent and heterogeneous movement of financial XML data.

This paper has multiple contributions. First, a type-2 fuzzy classification model has been proposed specifically for financial XML documents. This model clearly represents the uncertainty in tag sensitivity and produces a smooth and interpretable control surface for security-related attributes, based on recent insights into the role of FoU design in T2 FLS performance. Second, this classifier has been integrated with a selective XML cipher engine and experimentally compared the resulting scheme with full document encryption and a type-1 fuzzy partial encryption baseline. The evaluation shows that the proposed model can reduce encryption/decryption time and resource expenditure while maintaining strong confidentiality for the most critical elements, thus providing a more balanced and robust security solution for modern financial information systems.

The remainder of this research is organized as follows: Section 2 introduces some of the related work, Section 3 presents the main concepts of XML encryption and fuzzy logic technology, and Section 4 describes the meaning of fuzzy systems. The system model and design, and experimental results are presented in Sections 5 and 6, respectively. Finally, Section 7summarizes the conclusions.

2. Related works

Recent research has focused on integrating fuzzy logic (particularly Type 2) with encryption and security solutions to address the uncertainty inherent in IoT data and sensitive messages, supporting decisions about "when/what to encrypt" and "what level of protection we need." In this context, Chandnani and Verma presented a comparison of T2 FLS-based data encryption methods within IoT applications, highlighting Type 2's role in managing ambiguity and improving the reliability of security decisions compared to simpler approaches [3].

A recent review of functional encryption in IoT also presented advanced encryption trends (at the policy/attribute level) to reconcile confidentiality with sharing and access control requirements [7].

At the access control level (which is an integral part of any selective XML encryption in financial environments), Alshehri and colleagues proposed an attribute-based access control scheme for IoT using Hyperledger Fabric, demonstrating how policies and privileges are managed at a separate layer that can be linked to the decision to encrypt sensitive fields within messages [8].

In a similar vein to "enveloping security decisions" with fuzzy logic, Hosseinzadeh and colleagues presented a secure hierarchical routing scheme based on fuzzy logic to improve trust/security in networks, reflecting the ability of fuzzy logic to formulate security decisions under uncertainty [9].

Regarding "prioritizing security requirements" before building the encryption mechanism, Zohaib and colleagues used Analytic Hierarchy Process (fuzzy-AHP) to derive a ranked structure of IoT security success factors. This is practically important because financial XML encryption requires balancing confidentiality with performance and interoperability, especially when dealing with multiple use cases and threats [10].

More recent work has clearly moved towards context-based encryption and classification. Zeshan and colleagues proposed a context-aware encryption system based on a "fuzzy ontology" to classify IoT device/data information and prioritize confidentiality when applying the encryption algorithm. This is very similar to the idea of classifying financial XML fields/tags (sensitive/non-sensitive/medium) and then implementing element-level selective encryption [11].

Regarding "choosing the right encryption algorithm for limited devices," Radhakrishnan and colleagues evaluated the efficiency and security of lightweight encryption algorithms for resource-constrained IoT devices. This is a crucial element when designing selective XML encryption to avoid an unacceptable time burden when protecting fields [12].

A recent review in the Industrial Internet of Things (IIoT) environment security also provided a broad map of fuzzy logic applications in security, confirming the continuity of this research path through 2025 and reinforcing the legitimacy of integrating fuzzy logic with security mechanisms [13].

At the XML services threat level, Saeed and colleagues discussed security and authentication vulnerabilities in the Simple Object Access Protocol (SOAP) protocol and addressed XML-based attacks. This demonstrates practically that XML messages in business/services environments remain a live attack surface and require robust protection that goes beyond simply using full encryption [14].

In the 2025 trends that blend encryption with searchability/verification, Kuan Li presented a fuzzy cipher-based "search/verify" scheme with blockchain to improve privacy, search efficiency, and data integrity verification. This aligns with the need for verification and auditing in financial systems, even when parts of the message are encrypted [15].

Similarly, Huang et al. proposed an attribute-based encryption access control approach for high-dimensional medical data based on a fuzzy algorithm. This supports the idea of using fuzzy logic to evaluate data attributes/characteristics before enforcing encryption and access control [16].

Finally, recent work in authentication demonstrates that fuzzy logic is used not only in encryption but also in building more flexible authentication/verification layers. For example, a biometric authentication model for the IoT uses fuzzy logic to address biometric variability, which is important when linking financial XML encryption to the user/transaction context [17].

Although there are works that integrate fuzzy logic with encryption/policies, works that focus on context-aware encryption and data classification, and works that discuss XML risks in business services, integrating the classification of financial XML elements with T2FLS and then implementing selective encryption at the XML element level to achieve a tunable balance between confidentiality and performance remains an effective research area that deserves development.

3. Extensible markup language encryption and fuzzy logic technology

XML, known for its versatility and standardized syntax for general and financial messages, has been widely used by several financial institutions in their everyday operations. The extensive use of XML in financial transaction communications has sparked a collective interest in incorporating security measures into XML solutions to efficiently and securely protect exchanged XML files. Researchers have devised many approaches to protect XML data. Various approaches have been suggested at both the XML level [18, 19] and the network level to safeguard shared files. The W3C has enhanced the recommended models by offering standardized formats for the safe and reliable description of XML data. XML

key management, XML signature, and XML encryption were all established by the W3C [20]. The XML encryption standard delineates the methodology for encrypting an XML communication. This may need complete encryption of the message, selective partial encryption of certain segments, or the encryption of extraneous but relevant elements. This architecture can encrypt XML file, but there have been some performance and memory use problems; thus, there is still potential for development. However, financial institutions (such as banks) carry out numerous transactions every day, necessitating extensive XML encryption. Large volumes of files that are fully encrypted will cause performance and resource problems [12, 21].

Consequently, a technique for encrypting certain XML document chunks, a vocabulary for encoding encrypted bits, and processing rules for decrypting them are needed. A feature of XML encryption is element-wise encryption, a method for encrypting only certain parts of an XML document. To avoid performance or resource complications, a strategy must be devised to identify which segments of the XML document need real-time encryption. The components should be selected based on discerning criteria that identify sensitive information inside the XML document [22].

Here, sensitive sections inside each XML document can be distinguished using a fuzzy logic method [23]. A simple approach for drawing a definite conclusion from foggy, murky, erroneous, noisy, or missing input data is fuzzy logic. Fuzzy logic addresses control issues similarly to human decision-making processes, enabling swifter resolutions. Fuzzy logic offers a direct, rule-based <IF X AND Y THEN Z> approach for addressing control issues rather than attempting to formally delineate a system. The empirically based fuzzy logic model depends more on the operator's understanding of the system than on technical expertise. The fuzzy logic approach is evaluated with historical data and expert input. Fuzzy logic has been used by computer scientists for several years to integrate expert input into computational models for diverse applications. The advantage of the fuzzy approach lies in its capacity to manage variables with unclear interpretations and those whose relationships cannot be properly expressed. Fuzzy logic may use expert human judgment to define variables and their interrelations. The model may provide more accuracy and specificity about location compared to some other approaches [24].

4. Fuzzy systems

The fuzzy logic system includes several techniques, the most important of which are:

4.1. Mamdani

The Mamdani rule-based approach is the most renowned. It adheres to the following standards [25]:

a. All input values must be converted into fuzzy membership functions.
b. To calculate the fuzzy output functions, apply all necessary rules from the rule base.
c. To get "crisp" output values, defuzzify the fuzzy output functions.

4.2. Fuzzification

The procedure of allocating a system's numerical input to fuzzy sets with a certain degree of membership is termed fuzzyification. The degree of membership may range from 0 to 1. A value of 0 indicates that it does not belong to the fuzzy set, whereas a value of 1 signifies complete membership in the fuzzy set. Any value between 0 and 1 signifies the

degree of uncertainty about its inclusion in the collection. The system input may be used in a linguistically coherent manner by assigning it to these fuzzy sets, since they are often characterized by terminology [26].

For example, functions that correspond to a temperature scale are used to elucidate the meanings of the terms cold, warm, and hot in Fig. 1 below. Each of the three functions has three "truth values" at a given position on that scale. The three arrows (truth values) in the vertical line of the picture denote distinct temperatures. This temperature may be construed as "not hot" as the red arrow indicates zero, signifying the absence of members in the fuzzy set "hot." The orange arrow indicates "somewhat warm" at 0.2, while the blue arrow denotes "very cold" at 0.8. This temperature is classified inside the fuzzy sets "warm" and "cold" at ratios of 0.2 and 0.8, respectively. The degree of membership for each fuzzy set is established by fuzzification [27].

Each value has an increasing slope, attains a maximum at 1 (which may have a length of 0 or more), and then shows a decreasing slope; hence, fuzzy sets are sometimes described as triangular or trapezoidal curves [28]. A sigmoid function may also be used to characterize them. The standard logistic function is defined as [29]:

$$S(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

where x is the input value.
It has the symmetry characteristic listed below:

$$S(x) = S(-x) = 1 \tag{2}$$

From this, it is:

$$(S(x) + S(-x)) \cdot \left(S\left(y\right) + S\left(-y\right)\right) \cdot (S(z) + S(-z)) = 1 \tag{3}$$

where y is a real number input value.

5. System model and design

There is a set of primary components to the suggested model. Each component is an integral part of the whole model and has its own scope, acting as a standalone entity. XML file properties are divided into vital and non-critical parts using a variety of fuzzy classification algorithms, depending on how much they affect the file's security and functionality.

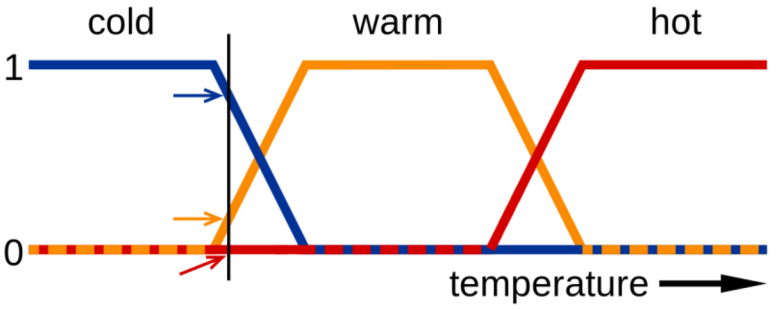

Fig. 1. Fuzzy logic example.

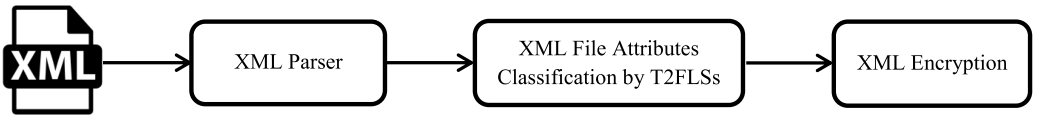

Fig. 2. The proposed model stages.

The main function of the fuzzy classification process is to determine the value of an attribute and assign it to the "Importance Level" XML tag, which is already there. The needed security level will be determined in the following stage using the assigned value. The following step entails applying object encoding to various XML file components. The entire XML file or just specific parts of it can be encrypted. Fig. 2 shows the stages of the proposed model.

5.1. Extensible markup language parser

There must be a stage in the model that is responsible for understanding the XML file and identifying the attributes in the text to facilitate the model's task of distinguishing and encoding those attributes. When XML file is entered into the proposed model, the model finds it challenging to understand the text of the file and cannot distinguish the attributes. XML file is used as the input for this stage, and the output is a list of attributes, which will be used as the input for the following stage, which is the categorization stage for attributes.

An example of an XML file that contains client information from a bank is shown below. The text contains the customer's name, the name of the bank, and information about the bank card, including the card number and expiration date.

```
<PaymentInfo xmlns="http://example.org/paymentv2" >
<Name>John Smith</Name>
<CreditCard Limit="5,000" Currency="USD">
<Number>4019 2445 0277 5567</Number>
<Issuer>Example Bank</Issuer>
<Expiration>04/02</Expiration>
</CreditCard>
</PaymentInfo>
```

5.2. Extensible markup language file attributes classification by Type-2 fuzzy logic

In this study, each security attribute, ConfidentialityImpact (CI), RegulatoryCriticality (RC), and AccessFrequency (AF), is modeled using Interval Type-2 Trapezoidal Membership Functions (IT2-TrMFs) with three linguistic terms: {Low, Medium, High}. An IT2-TrMF is defined by an Upper Membership Function (UMF) and a Lower Membership Function (LMF), whose bounded region forms the FoU, allowing the model to represent uncertainty in tag sensitivity more explicitly than type-1 fuzzy sets.

For each term Linguistic Label Level (LLL), the UMF and LMF are trapezoids parameterized as (aU,bU,cU,dU)(a_U,b_U,c_U,d_U)(aU,bU,cU,dU) and (aL,bL,cL,dL)(a_L,b_L,c_L,d_L)(aL,bL,cL,dL), where the LMF is typically narrower (more conservative) to encode uncertainty through the FoU width.

UMF includes:

- The upper bound of the Type-2 fuzzy set membership grades.
- Usually a trapezoidal function parameterized as (aU, bU, cU, dU).

LMF includes:

- The lower bound, typically narrower than the UMF, representing more conservative membership values.
- Parameterized as (aL, bL, cL, dL).

The distance between UMF and LMF defines the FoU, capturing the uncertainty in assigning a precise membership value to a linguistic label.

The fuzzy inference follows the standard Interval Type-2 Fuzzy Logic System (IT2FLS) pipeline (fuzzification → rule firing with interval strengths → aggregation → Karnik–Mendel type-reduction → defuzzification) to obtain a crisp TagImportance (TI) score ∈ [0, 1].

The core of the proposed intelligent encryption model is an interval T2FLS that maps a set of security-related attributes of each XML element into a continuous importance score used to drive selective encryption decisions.

Conceptually, the T2FLS consists of four main components: (i) a fuzzifier and membership functions, (ii) a fuzzy rule base, (iii) a type-2 inference engine, and (iv) type-reduction and defuzzification to obtain a crisp output [30, 31].

To address the aforementioned objection of type-1 fuzzy sets, ambiguity about the membership function may be included in type-2 fuzzy sets. Furthermore, in the absence of uncertainty, a type-2 fuzzy set reduces to a type-1 fuzzy set, analogous to the transition from probability to determinism when unpredictability is eliminated. Type-1 fuzzy systems use a static membership function, whereas type-2 fuzzy systems utilize a dynamic membership function. The transformation of input values into fuzzy variables is dictated by a fuzzy set [32].

Fuzzification determines the extent to which the crisp inputs x and y are a part of the fuzzy set and where they fall within it. The transaction currency is one of the factors that is represented by a linguistic variable. The range of transaction amounts is depicted on the x-axis. The magnitude of each value in the linguistic descriptor is shown on the y-axis. Rule assessment uses the qualified fuzzy rules in conjunction with the fuzzy inputs. In cases of uncertainty, fuzzy operators (AND / OR) are used to get a singular value.

Fig. 3 shows that the model chose important attributes, the importance of which lies in the security of the data it contains, which is the bank card number and its expiration date. The “Truth Value” outcome value will be used with the membership function to evaluate rules.

Scaled rules are amalgamated into a singular fuzzy set for each variable using the aggregation of the rule outputs process. The value attributed to each tag in the output must be unique and succinct.

Fig. 4 illustrates the control surface of a Type-2 Fuzzy Inference System (FIS) used for classification purposes. The control surface provides a three-dimensional representation of the relationship between two input variables and the system’s output, which represents the final classification decision.

The horizontal axes of the control surface represent the input variables. Each variable is modeled using type-2 fuzzy sets characterized by the presence of a higher-order affiliation function (UMF) and a lower-order affiliation function (LMF). This structure allows the system to explicitly represent the uncertainty in the input data and grammar, which is a major advantage of T2FLS compared to T1FLS.

The vertical axis represents the final output value (the explicit value) obtained after performing type reduction and defuzzification operations. The smoothness and fluidity of the control surface reflect the ability of the type 2 FIS to produce gradual output transitions

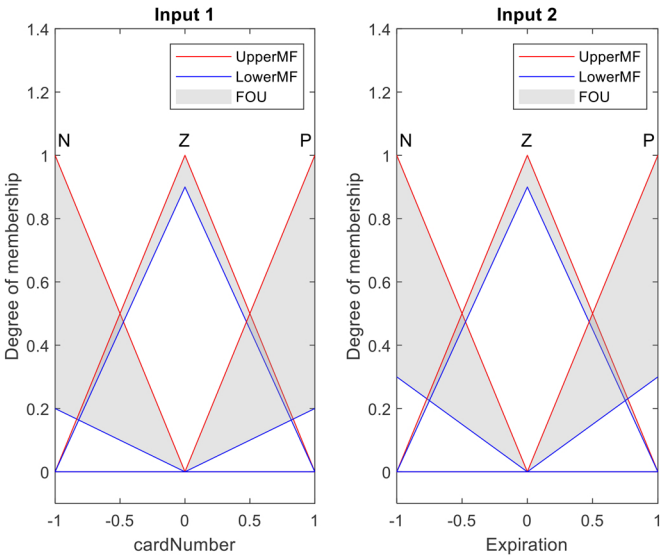

Fig. 3. Input attributes for the step of T2FLS.

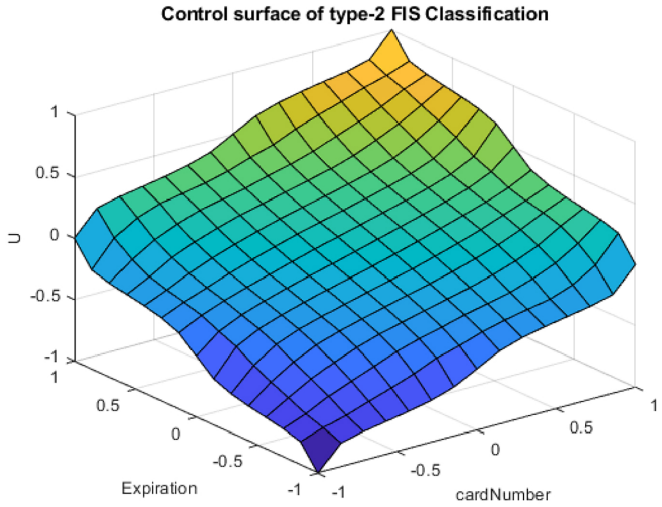

Fig. 4. Control surface of type-2 FIS classification.

in response to changes in input variables. This reduces sharp edges at the decision boundary and enhances classification robustness under uncertainty.

Compared to the control surfaces of T1FLS, the type 2 control surface exhibits the FoU effect embedded within the membership functions. This improves the system's resistance to noise, input inaccuracies, and rule uncertainties, which is crucial in real-world classification applications characterized by variable and fluctuating data.

Overall, Fig. 4 demonstrates that the type 2 FIS can represent nonlinear input-output relationships while maintaining stability and reliability, making it suitable for complex decision-making and classification applications.

The algorithm presented below aims to assess the "importance of a tag" within an XML file and generate a numerical value (TI) within the range [0, 1] representing the

Algorithm 1 Interval Type-2 Fuzzy Tag Importance Evaluation.

Input: XML element e
Output: TagImportance score TI ∈ [0, 1]
Extract crisp attributes from e
  CI ← ComputeConfidentialityImpact(e)
  RC ← ComputeRegulatoryCriticality(e)
  AF ← ComputeAccessFrequency(e)
Normalize CI, RC, AF to [0, 1]
Fuzzification
  For each linguistic term L ∈ {Low, Medium, High}
    compute upper/lower MFs $\mu\bar{L}(x)$, $\mu\underline{L}(x)$
    obtain interval memberships for $\overline{CI}$, $\overline{RC}$, AF
For each rule Rk in the rule base
  compute interval firing strength $\mu\tilde{R}k$
  obtain interval Type-2 output set $\hat{Y}k$
Aggregate all $\hat{Y}k$ to form the global output set $\hat{Y}$
Apply KM type-reduction on $\hat{Y} \rightarrow$ [yL, yR]
Defuzzification
  TI ← (yL + yR) / 2
return TI

sensitivity/importance of that tag. This allows for subsequent security decisions, such as: should this tag be encrypted or not? (or what level of protection is required). The algorithm relies on an IT2FLS because it is better able to represent uncertainty than T1FLS, especially when sensitivity or compliance estimates are imprecise or vary depending on the context.

5.3. Extensible markup language encryption

XML encryption is a standard established by the W3C in 2002. It includes directives for encrypting and decrypting data, together with the syntax for representing encrypted data in XML. It also encompasses a compilation of encryption methods, including triple Data Encryption Standard (DES), AES, and Rivest-Shamir-Adleman algorithm (RSA). Any data, including an XML document, may be encrypted using XML, including XML elements, their content, and arbitrary data. Examine the situation in which it is necessary to encrypt the <CreditCard> element from the subsequent XML file.

An example of a portion of an XML transaction snippet describing a customer's payment information is shown below:

<PaymentInfo xmlns="http://example.org/paymentv2'' >
<Name>John Smith</Name>
<CreditCard Limit="5,000" Currency="USD">
<Number>4019 2445 0277 5567</Number>
<Issuer>Example Bank</Issuer>
<Expiration>04/02</Expiration>
</CreditCard>
</PaymentInfo>

In above example, it can be notice that:

The '<PaymentInfo>' element is the root (in this snippet) and collects the payment data within a specific namespace (paymentv2) to avoid tag conflicts with other XML schemes.

The '<Name>' element contains the transaction owner's name.

The ‘<CreditCard>‘ element represents a credit card and contains attributes such as:

– ‘Limit=“5,000”: The credit limit.
– ‘Currency=“USD”: The currency of the credit limit.

Within ‘<CreditCard>‘ are sensitive sub-elements:

– ‘<Number>‘: The card number (highly sensitive data and considered PCI DSS data).
– ‘<Issuer>‘: The card issuer/bank.
– ‘<Expiration>‘: The card’s expiration date.

In the context of this research (selective encryption), <Number> and possibly <Expiration> are usually encrypted because they are the most sensitive, while <Issuer> and the limit/currency properties can be left unencrypted if the policy allows it to reduce the overhead while maintaining confidentiality.

The encrypted <CreditCard> element is represented by the “EncryptedData” element in the following XML language. The triple DES encryption algorithm is described by the <EncryptionMethod> element in this example. In this example, the decryption key retrieval information is contained in the <KeyInfo> element, which is a <KeyName> element. The ciphertext acquired by serializing and encrypting the <CreditCard> element is contained in the <CipherValue> element.

<PaymentInfo xmlns=’http://example.org/paymentv2’>
<Name>John Smith</Name>
<EncryptedData Type=’http://www.w3.org/2001/04/xmlenc#Element’
xmlns=’http://www.w3.org/2001/04/xmlenc#’>
<EncryptionMethod
Algorithm=’http://www.w3.org/2001/04/xmlenc#tripledes-cbc’/>
<KeyInfo xmlns=’http://www.w3.org/2000/09/xmldsig#’>
<KeyName>John Smith</KeyName>
</KeyInfo>
<CipherData>
<CipherValue>ydUNqHkMrD...</CipherValue>
</CipherData>
</EncryptedData>
</PaymentInfo>

It is assumed in the above XML content that the sender and recipient have a secret key. The <CreditCard> part can be encrypted if the recipient possesses a public and private key pair, which is most usually the case. The <EncryptedData> element is identical to that element. However, an <EncryptedKey> is present in the <KeyInfo> element.

6. Experimental results

6.1. Experimental setup

To evaluate the effectiveness of the proposed T2FLS–based XML encryption model, a set of experiments was conducted. The dataset consisted of 100 XML files representing simulated financial transactions, with file sizes ranging from 10 KB to 5 MB [33, 34].

All experiments were performed on a computer with the following specifications: Windows 11 (64-bit), Intel Core i7 processor with 3.2 GHz, and 16 GB RAM. Python 3.11 with the pycryptodome library has been used for programming the proposed model.

Table 1. Performance comparison of encryption models.

Metric	Full Encryption	T1FLS + Partial	The Proposed Model
Average Encryption Time (ms)	1250	880	640
Average Decryption Time (ms)	1180	860	615
File Size Overhead (%)	42%	28%	17%
Memory Usage (MB)	520	420	350

6.2. Comparison with the baseline methods

The proposed model is compared with: Eq. (1) full XML encryption (baseline – AES 256 applied to the entire file) and Eq. (2) type-1 fuzzy + partial encryption (selective encryption using type-1 fuzzy classification).

The performance metrics evaluated were: Eq. (1) average encryption time (ms), Eq. (2) average decryption time (ms), Eq. (3) file size overhead (%), and (4) memory usage (MB).

Table 1 shows the performance comparison of encryption models.

The proposed model reduced the encryption time to ~ 49% compared to full encryption, and ~ 27% compared to T1FLS + partial encryption. The proposed model got the lowest overhead (17%), making it more efficient for major financial transactions. The proposed model consumed ~ 32% less memory compared to full encryption, only due to the selective processing of sensitive tags.

These results confirm that the proposed model successfully balances security and efficiency, making it suitable for real-time financial applications where both privacy and performance are important.

6.3. Security analysis

To ensure privacy, integrity and availability of economic XML transactions, the resistance of the model is necessary to evaluate the resistance of the model. In this section, the safety of the proposed model has been analyzed compared to full XML encryption and T1 FLS encryption through: Eq. (1) threat model, Eq. (2) security features of the proposed model, and Eq. (3) comparative analysis of related works.

6.3.1. Threat model

The following security attacks were considered:

a. Brute force attack: An attempt to find the reverse key to break the encryption.
b. Repley attack: Encrypted XML messages were re-embedded to gain unauthorized access.
c. XML wrapping attack: Manipulation of the structure of XML documents to circumvent certification.
d. Privacy leakage: Exposure to sensitive data due to incomplete or weak encryption.

6.3.2. Security features of the proposed model

The proposed model has the following security features:

a. Strong encryption algorithm: Using the AES with 128–256-bit keys provides strong resistance against brute-force key-search attacks (AES supports 128/192/256-bit keys).
b. Selective tag level encryption: Only sensitive fields (e.g. <CreditCard>, <Puplirya-date>) are encrypted, at least exposure.

Table 2. Comparative security analysis.

Attack Type	Full Encryption	T1FLS + Partial	The Proposed Model
Brute Force Attack	Strong (AES-256)	Strong (AES-128/256)	Strong (AES-128/256)
Replay Attack	Vulnerable (if no nonce/timestamp)	Partially Secure (static rules)	Secure (dynamic rules + nonce/timestamp)
XML Wrapping Attack	Vulnerable (structure unchanged)	Partially Secure	Secure (tag-level dynamic validation)
Confidentiality Leakage	None (full file encrypted)	Moderate (some sensitive tags may be misclassified)	Low (adaptive T2FLS ensures accurate sensitivity detection)

c. T2FLS classifier: Provides dynamic identity of sensitive tags, and prevents the attackers from estimating which fields are encrypted.
d. Nonce and timestamp integration: Adding a nonce (a unique "number used once") and a timestamp to each message helps prevent replay attacks by enabling the receiver to verify freshness (rejecting duplicated nonces and messages outside an acceptable time window).
e. XML signature compatibility: Integrity supports W3C standards for integrity verification and safety against XML raping.

Table 2 shows the comparative security analysis.

All models provide strong protection due to AES encryption. However, the proposed model gets efficiency without compromising on the strength. The use of nonce + time stamp in the proposed model increases safety compared to static T1FLS encryption. Dynamic classification and tag-level encryption in the proposed model make it more flexible for structural manipulation. T2FLS reduces the possibility of leaving sensitive areas unknown compared to unclear logic T1FLS.

The proposed model not only reduces computational overhead but also increases the safety of sophisticated XML-specific attacks. This makes it very suitable for economic and e-commerce systems where both efficiency and strong safety are needed.

6.3.3. Comparison with related works

The proposed model, in theory, is an improvement over traditional decision-making systems based on T2FLS. However, the empirical evaluation in this research deliberately focuses on two commonly used and practically relevant baselines: Eq. (1) full XML encryption according to the recommendations of the W3C XML cryptography standard, and Eq. (2) partial encryption based on T1FLS, which represents the prevailing smart selective encryption strategy in previous banking and financial applications.

To complement this quantitative comparison, Table 3 presents a qualitative analysis of representative works using either T1FLS, T2FLS ontology, or standard XML encryption mechanisms. This table highlights that the proposed model is, to the best of our knowledge, the first T2FLS-driven smart XML encryption model specifically designed for financial transactions, while maintaining compliance with W3C encryption and signature standards.

Furthermore, the evaluation can be strengthened by incorporating additional, more robust baselines: Eq. (1) an alternative encryption method that explicitly uses traditional T2FLS, and Eq. (2) one or more recently published high-performance encryption models in the same application area. Including such stronger benchmarks will make performance comparisons more accurate and will more objectively highlight the actual improvements achieved by the proposed model.

Therefore, establishing a more comprehensive numerical benchmark for comparison with T2FLS models and modern high-performance encryption models is an important

Table 3. Comparative analysis with related works.

Authors	Contribution	Strengths	Weaknesses
Chandnani and Verma (2023)	Evaluation/comparison of T2FLS used with encryption techniques for IoT data protection.	Focuses on T2FLS in a security context and provides a useful "model comparison" perspective for selecting/justifying T2FLS.	Its scope is IoT, not financial XML, and it does not discuss selective encryption at the XML tag level or XML (wrapping/replay) attacks within transactional message structures.
Shahzad et al. (2022)	Survey on functional encryption in IoT, its applications, and related primitives.	Strengths of "broad coverage": Maps policy/function-based cryptographic ideas in IoT.	Because it is a literature review, it does not present a selective XML encryption model or time/overhead/memory results on transactional XML files.
Alshehri et al. (2022)	Attribute-Based Access Control (ABAC) scheme for IoT access control using Hyperledger Fabric.	Strengths of policy/access control-level processing and a blockchain architecture suitable for privilege tracking.	It does not focus on XML structure encryption or "choosing fields/tags" to encrypt within an XML message; rather, it focuses on who has access.
Hosseinzadeh et al. (2023)	Secure hierarchical routing scheme for the IoT using the fuzzy trust framework and firefly algorithm.	Strengths of "network security/trust": Adds a trust layer and secure routing suitable for sensitive environments such as IoT-healthcare.	Its goal is secure routing, not XML encryption, protecting financial XML transactions, or reducing XML encryption overhead.
Zohaib et al. (2024)	Identifying and categorizing 21 success factors for IoT security, then ranking them using Fuzzy-AHP to guide managerial/technical decision-making.	Strengths of "decision routing": offers taxonomy + prioritization useful for justifying enterprise/system-level security choices.	It is not an encryption model or implementation mechanism for selective XML encryption/overhead measurement; it focuses on factors and their weighting.
Zeshan et al. (2024)	A context-aware encryption system in IoT that uses fuzzy ontology and takes into account device capabilities and security priorities when selecting protection.	Strengths of "decision context": Combines ontology with fuzzy logic for capability/priority-based cryptographic selection.	Its scope is IoT context-aware and not directly aligned with W3C XML encryption/signature, nor does it explicitly target XML financial tag encryption and XML threats such as wrapping.
The proposed model	Interval Type-2 Fuzzy Tag Importance algorithm for calculating TI $\in$ [0, 1] based on CI/RC/AF, followed by Eq. (2) selective tag-level encryption within XML, with Eq. (3) nonce/timestamp and (4) XML signature compatibility, and evaluation of practical performance.	Strengths of intelligent decision integration + XML standardization: classifies tag sensitivity and then selectively encrypts it to reduce cost while preserving AES.	It relies on a baseline/affiliation function design and requires strengthening the comparison with more recent references/universal data standards (mentioned as a limitation/future work).

direction for future work. This research requires standardized datasets and publicly available applications to ensure a fair comparison of performance and security characteristics across heterogeneous environments, and we explicitly acknowledge that this is a current limitation of the evaluation.

Recent work in IoT security shows a clear trend toward integrating fuzzy logic with security/control mechanisms, but often outside the realm of financial XML transactions. For example, Chandnani and Verma [3] focused on comparing T2FLS-based IoT data encryption techniques to illustrate the differences between T2FLS models as a security improvement approach.

Meanwhile, Shahzad et al. [8] conducted a survey of functional encryption in IoT from the perspective of cryptographic primitives and access control applications. This survey is robust in terms of literature coverage but lacks an implementation model geared toward XML or selective in-message encryption. Similarly, Alshehri et al.'s [9] suggested Adaptive Attribute-based Access Control (AAC)-IoT targeted access control (ABAC) via Hyperledger Fabric to address access privileges in IoT, meaning it protects "who accesses" rather than "how XML content is encrypted internally." Hosseinzadeh et al. [10] presented a secure hierarchical routing scheme based on a fuzzy trust framework and the firefly algorithm for healthy IoT networks. This scheme is robust at the network/routing level but is not geared towards encrypting XML documents or messages.

Zohaib et al. [11] presented a structured analysis of IoT security success factors using fuzzy-AHP to prioritize at the administrative/governance decision level. Meanwhile, Zeshan et al. [12] proposed context-aware encryption via fuzzy ontology that considers device capabilities and confidentiality priorities. This approach is closer to the concept of "smart decision before encryption", but remains more context-specific to IoT than a standardized, financially compliant "selective XML encryption" model.

In contrast, this research contributes to "determining what to encrypt within financial XML" by using IT2FLS to generate a tag importance score of $TI \in [0, 1]$ based on clear security attributes such as CI, RC, and AF, and then applying tag-level selective encryption. This research also incorporates features resistant to XML message-related attacks such as replay via nonce/timestamp and supports integrity through compatibility with XML signature.

7. Conclusion

In this research, an intelligent XML encryption model based on T2FLS unclear logic has been proposed to increase the safety of economic XML transactions. Unlike traditional complete encryption methods, which impose important calculation and storage costs, the proposed model creeps selectively on the most sensitive elements identified through T2FLS unclear classification. Experimental results demonstrated that the proposed model reduced encryption and decryption time by approximately 49%, compared to complete encryption and 27% T1FLS encryption. In addition, the file size overhead was reduced to 17%, and memory use was reduced to 32%, which confirmed the efficiency of the proposed model.

From the security perspective, the proposed model showed strength against general attacks, including brutal forces, reprises and XML raping attacks. The use of T2FLS unclear logic allowed more accurate detection of sensitive tags, which reduced the risk of privacy leak compared to T1FLS unclear model. These findings highlight the ability of the proposed model to provide a balanced solution that combines both strong safety and high performance, making it suitable for real-time economic and e-commerce systems.

Despite these promising results, many restrictions remain. The proposed model still depends on the unclear rules, which can introduce degeneracy into sensitivity classification. In addition, while experiments used simulated economic data sets, the proposed model requires further verification with real XML bank and business systems to confirm the generality of the model.

Many directions can be used in future to extend this research, such as:

a. By integrating monitored machine learning techniques, the unclear ruling can be automatically generated by enabling dynamic adaptation without expert intervention.
b. Expand the proposed model to operate in blockchain-based economic systems, and use account technology distributed for further integrity guarantee.
c. Include mechanisms for detecting real-time deviations to identify fraud or malicious XML transactions with encryption.
d. Evaluate the scalability of the model in the cloud and IoT environment, where XML remains a widely used data exchange standard.

Acknowledgment

None.

Authors contributions

Faiez Musa Lahmood AlRufaye**:** Responsible for writing, review, and analytical editing; also contributed to outlining future research directions**.** Seham Ahmed Hashem: Review and editorial process.

Conflict of interest

Regarding the publishing of this paper, the authors state that they have no conflicts of interest.

Data availability

The dataset was synthetically generated for this study and is not publicly available. Details of the data generation process are provided in the experimental setup subsection.

References

1. D. Eastlake, “Additional XML security uniform resource identifiers,” *RFC 9231*, Jul. 2022, doi: 10.17487/RFC9231.
2. T. S. Almeida, A. D. S. Mendes, P. M. S. R. Rizol, and M. A. G. Machado, “Performance analysis of interval type-2 fuzzy $\bar{X}$ and R control charts,” *Applied Sciences*, vol. 13, no. 20, Oct. 2023, Art. no. 11594, doi: 10.3390/app132011594.
3. N. Chandnani and K. Verma, “A comparison on type-II fuzzy logic based data encryption techniques in the internet of things,” *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 44, no. 2, pp. 2109–2116, Oct. 2023, doi: 10.3233/JIFS-220570.
4. Z. Janková and E. Rakovská, “Comparison uncertainty of different types of membership functions in T2FLS: Case of international financial market,” *Applied Sciences*, vol. 12, no. 2, Jan. 2022, Art. no. 918, doi: 10.3390/app12020918.

5. R. Raj and J.-M. Yang, "Analytical structure and performance of interval type-2 fuzzy two-term controllers with varying footprint of uncertainty," *International Journal of Computational Intelligence Systems*, vol. 15, no. 1, Dec. 2022, Art. no. 106, doi: 10.1007/s44196-022-00162-w.
6. L. Zhu, J. Wang, and L. Bai, "A general characterization of integrating and querying heterogeneous fuzzy spatiotemporal XML data," *Earth Science Informatics*, vol. 16, no. 4, pp. 3303–3321, Sep. 2023, doi: 10.1007/s12145-023-01091-8.
7. K. Shahzad, T. Zia, and E-U.-H. Qazi, "A review of functional encryption in IoT applications," *Sensors*, vol. 22, no. 19, Oct. 2022, Art. no. 7567, doi: 10.3390/s22197567.
8. M. Alshehri *et al.*, "AAC-IoT: Attribute access control scheme for IoT using Hyperledger Fabric," *Applied Sciences*, vol. 12, no. 16, Aug. 2022, Art. no. 8111, doi: 10.3390/app12168111.
9. A. Hosseinzadeh *et al.*, "A fuzzy logic-based secure hierarchical routing scheme for wireless sensor networks," *Scientific Reports*, vol. 13, no. 1, 2023, Art. no. 11058, doi: 10.1038/s41598-023-38203-9.
10. M. Zohaib, A. A. Alsanad, and M. A. Akbar, "Success factors of IoT security: A structured analysis using fuzzy-AHP," *IEEE Access*, vol. 12, pp. 186186–186209, 2024, doi: 10.1109/ACCESS.2024.3464102.
11. F. Zeshan, Z. Dar, A. Ahmad, and T. Malik, "A fuzzy ontology-based context-aware encryption approach in IoT through device and information classification," *The Journal of Supercomputing*, vol. 80, no. 16, pp. 23311–23356, Jul. 2024, doi: 10.1007/s11227-024-06317-0.
12. I. Radhakrishnan, S. Jadon, and P. B. Honnavalli, "Efficiency and security evaluation of lightweight cryptographic algorithms for resource-constrained IoT devices," *Sensors*, vol. 24, no. 12, Jun. 2024, Art. no. 4008, doi: 10.3390/s24124008.
13. E. Krzysztoń, D. Mikolajewski, and P. Prokopowicz, "Review of fuzzy methods application in IIoT security—challenges and perspectives," *Electronics*, vol. 14, no. 17, Aug. 2025, Art. no. 3475, doi: 10.3390/electronics14173475.
14. M. M Saeed, "Cloud security and authentication vulnerabilities in SOAP protocol: Addressing XML-based attacks," *Frontiers in Computer Science*, vol. 7, Sep. 2025, Art. no. 1595624, doi: 10.3389/fcomp.2025.1595624.
15. K. Li, "Fuzzy encryption search scheme and data verification mechanism based on blockchain," *International Journal of Computational Intelligence Systems*, vol. 18, no. 1, Apr. 2025, Art. no. 93, doi: 10.1007/s44196-025-00804-9.
16. Y. Huang, T. Teng, Y. Li, and M. Zhang, "Attribute encryption access control method of high dimensional medical data based on fuzzy algorithm," *PLOS ONE*, vol. 20, no. 3, 2025, Art. no. e0317119, doi: 10.1371/journal.pone.0317119.
17. G. M. El-Banby, S. M. El-Gazar, W. El-Shafai, R. M. Ghoniem, and F. E. Abd El-Samie, "Fuzzy-logic-based biometric authentication for IoT access using speech and ECG signals," *Traitement du Signal*, vol. 42, no. 5, pp. 2539–2547, Oct. 2025, doi: 10.18280/ts.420508.
18. L. Bai and L. Zhu, "Transformation of fuzzy spatiotemporal data between XML and object-oriented databases," in *Fuzzy Spatiotemporal XML Data Management*, *Studies in Computational Intelligence*. Switzerland: Springer, 2025, ch. 3, pp. 77–113, doi: 10.1007/978-3-031-81033-6_3.
19. C. Peraza, P. Ochoa, O. Castillo, and P. Melin, "Behavioral analysis of an interval type-2 fuzzy controller designed with harmony search enhanced with shadowed type-2 fuzzy parameter adaptation," *Applied Sciences*, vol. 13, no. 13, Jul. 2023, Art. no. 7964, doi: 10.3390/app13137964.
20. L. Guo and H. Wu, "An XML privacy-preserving data disclosure decision scheme," *Security and Communication Networks*, vol. 2022, Feb. 2022, Art. no. 9099722, doi: 10.1155/2022/9099722.
21. I. Riali, M. Fareh, and F. Bobillo, "ProbFuzzOnto: A fuzzy ontology-driven uncertainty approach using fuzzy Bayesian networks," *International Journal of Fuzzy Systems*, vol. 27, no. 3, pp. 680–700, Oct. 2024, doi: 10.1007/s40815-024-01796-y.
22. Z. Ma and L. Yan, "Data modeling and querying with fuzzy sets: A systematic survey," *Fuzzy Sets and Systems*, vol. 445, pp. 147–183, Sep. 2022, doi: 10.1016/j.fss.2022.01.006.
23. M. Parciak, B. Vandevoort, F. Neven, L. M. Peeters, and S. Vansummeren, "Schema matching with large language models: An experimental study," 2024, *arXiv:2407.11852*.
24. ISO 20022 Registration Authority, "ISO 20022 message definitions," ISO 20022. Accessed: 15 May 2025. [Online] Available: https://www.iso20022.org/iso-20022-message-definitions.
25. Goldman Sachs Developer, "pain.008.001.02 sample file," Transaction Banking Developer Documentation. Accessed: 15 May 2025. [Online] Available: https://developer.gs.com/docs/services/transaction-banking/pain008-sample-file.
26. M. Al-atar and A. Sali, "Approximate integrity constraints in incomplete databases with limited domains," *Annals of Mathematics and Artificial Intelligence*, vol. 39, no. 5, pp. 759–786, Feb. 2025, doi: 10.1007/s10472-025-09967-9.
27. M. Grohe and P. Lindner, "Independence in infinite probabilistic databases," *Journal of the ACM*, vol. 69, no. 5, Oct. 2022, Art. no. 37, doi: 10.1145/3549525.

28. S. Maniu and P. Senellart, “Database theory in action: Making provenance and probabilistic database theory work in practice (invited talk),” in *Proc. 28th Int. Conf. on Database Theory (ICDT 2025)*, Barcelona, Spain, pp. 1–6, doi: 10.4230/LIPIcs.ICDT.2025.33.
29. A. Gilad, A. Imber, and B. Kimelfeld, “The consistency of probabilistic databases with independent cells,” in *Proc. 26th Int. Conf. on Database Theory (ICDT 2023)*, Ioannina, Greece, pp. 1–19, doi: 10.4230/LIPIcs.ICDT.2023.22.
30. Z. Guo *et al.*, “A survey on uncertainty reasoning and quantification in belief theory and its application to deep learning,” *Information Fusion*, vol. 101, Jan. 2024, Art. no. 101987, doi: 10.1016/j.inffus.2023.101987.
31. J. Sunkavalli, R. H. Lalitha, R. Reenadevi, M. Dhivya, and K. Sreeramamurthy, “Ontology-based multi-agent system on fuzzy markup language in healthy lifestyle,” *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 21s, pp. 1–10, Mar. 2024.
32. V. K. Ahlawat, G. Agarwal, V. Goel, K. L. Hui, and M. Sain, “A novel encrypted XML streaming technique for indexing data on multiple channels,” *KSII Transactions on Internet and Information Systems*, vol. 18, no. 7, pp. 1840–1867, Jul. 2024, doi: 10.3837/tiis.2024.07.007.
33. F. Valdez, O. Castillo, and P. Melin, “A bibliometric review of type-3 fuzzy logic applications,” *Mathematics*, vol. 13, no. 3, Jan. 2025, Art. no. 375, doi: 10.3390/math13030375.
34. Y. S. Al-Nahhas, L. A. Hadidi, M. S. Islam, M. Skitmore, and Z. Abunada, “Modified Mamdani-fuzzy inference system for predicting the cost overrun of construction projects,” *Applied Soft Computing*, vol. 151, Jan. 2024, Art. no. 111152, doi: 10.1016/j.asoc.2023.111152.