A Fuzzy Inference System (FIS) to Evaluate the Security Readiness of Cloud Service Providers

1st Ana Valente (38.5%)
93307
Master in Mathematics and Applications
Department of Mathematics
University of Aveiro

2nd Beatriz Mesquita (38.5%)
115367
Master in Data Science
Department of Mathematics
University of Aveiro

3rd Obafemi Raymondjoy (23%)
111949
Master in Mathematics and Applications
Department of Mathematics
University of Aveiro

***Abstract*—Fuzzy logic can be characterized as many-valued logic with unique properties aiming at modeling the vagueness phenomenon and some parts of the meaning of natural language via a graded approach, considering that the truth value of variables may be any real number ranging from 0 to 1. [16] One of the biggest concerns that still arise from cloud service users is trust in cloud service providers, mainly due to security and privacy issues. In order to make an assessment of cloud service users' trust in cloud service providers its applied a fuzzy logic approach. Several users were asked to evaluate cloud service providers they have already used, obtaining, from a fuzzy inference system (FIS), a security index for it. This evaluation allows the user to determine which cloud service provider is the most trustworthy and also enables a self-assessment by the cloud service providers, as it works as feedback from users.**

***Index Terms*—fuzzy logic, cloud service users, cloud service providers, FIS, security index**

I. Introduction

A cloud service provider is a third-party company offering a cloud-based platform, infrastructure, application, or storage services [8]. Cloud service providers use their own data centers and compute resources to host cloud computing-based infrastructure and platform services for customer organizations [2].

Besides the pay-per-use model, cloud service users can take advantage of scalability and flexibility by not being limited to physical constraints of on-premises servers, the reliability of multiple data centers with multiple redundancies, customization by configuring servers to your preferences [8], there is also the benefit of mobility, since resources and services purchased from a cloud service provider can be accessed from any physical location that has a working network connection, and disaster recovery [2]. Despite the numerous perks, cloud service users face a trust deficit in cloud service providers when trusting critical data, organizations risk security breaches, compromised credentials, and other substantial security risks. These security and privacy aspects are major contributors for the hindrance of adhesion to cloud computing. Therefore, the development of a model which reflects the assessment of cloud service providers, as well as quantifies trust level is of utmost importance to overcome this trust deficit between cloud service users and providers [19].

This study aims to analyze and evaluate the security readiness of cloud service providers based on a fuzzy logic approach that allows the determination of their trustworthiness. Based on an article [19] on the matter, we develop a fuzzy inference system that yields a quantitative security index to cloud service providers.

II. Methodology

A. Description of the theoretical FIS

Fuzzy logic is an approach to variable processing that allows for multiple possible truth values to be processed through the same variable.

Whereas classical logic deals with statements of absolute truth, fuzzy logic addresses sets with subjective or relative definitions. This method resembles human reasoning, in a way that relies on vague or imprecise values rather than absolute truth or falsehood. [4]

Fuzzy logic derives from fuzzy set theory, based on which it is possible to define a system that provides answers to many questions and copes with unreliable and incomplete information; this system is called **fuzzy inference system** (FIS). There are two different types of fuzzy inference systems, the Mamdani and Takagi-Sugeno. Although Takagi-Sugeno FIS is more computationally efficient, Mamdani offers a more interpretable rule base, being more intuitive and well-suited to human input. Considering these aspects, the Mamdani FIS is chosen over Takagi-Sugeno for this application. A fuzzy inference system architecture is composed by rules of inference, process of fuzzification, intelligence/inference engine and process of defuzzification.

The process of fuzzification consists in the transformation of crisp values, defined as values that are not fuzzy, into fuzzy values. Specifically, the crisp data is mapped to a fuzzy set which contains the membership functions and linguistic values. According to fuzzy set theory, a fuzzy set is defined as any set that allows its elements to have different degree of membership, denominated as membership function, defined

in the interval $[0, 1]$. Let U be the universe of discourse, $X = \{x_1, x_2, \ldots, x_n\} \subseteq U$ and $X \subseteq [0, 1]$, if A is a fuzzy set then

$$A = \{(x, \mu_A(x)) \mid \mu_A : x \to [0, 1]\}$$

where x is an element of set X, $\mu_A(x)$ is the membership function of x in A and μ_A is the membership degree of x in A.

In this study, it is considered a triangular membership function defined as

$$\mu(x; a, b, c) = \begin{cases} 1 & \text{if } x < a \\ \frac{x-a}{b-a} & \text{if } a \leq x < b \\ \frac{c-x}{c-b} & \text{if } b \leq x < c \\ 0 & \text{if } c \leq x \end{cases}$$

or, more compactly,

$$\mu(x; a, b, c) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right)$$

where x is an element of set X and a, b and c are parameters of the membership function.

Once the process of fuzzification is done, the fuzzy input is then evaluated through an inference engine, which is responsible for applying the inference rules, represented by a set of conditional statements and linguistic variables, to the fuzzy input, in order to generate the fuzzy output.

When formulating the inference rules to use them in the context of fuzzy sets it is mandatory to employ fuzzy sets operators. The whole class of functions that are called T-norms can be used as fuzzy intersections, while S-norms (T-co-norms) can be used as fuzzy unions. In this study, the intersection of fuzzy sets is defined as $T(a, b) = \min(a, b)$ and the union of fuzzy sets is given by $S(a, b) = \max(a, b)$ for any $a, b \in [0, 1]$. In this study, there is also an exploration of Łukasiewicz operators, defined as $T(a, b) = \max(0, a + b - 1)$ and $S(a, b) = \min(1, a + b)$, and Gödel operators, also known as product operators, described by $T(a, b) = a \cdot b$ and $S(a, b) = a + b - a \cdot b$, such that $a, b \in [0, 1]$.

The defuzzification process is applied in order to make better analysis and interpretation considering the problem context. In defuzzification, the fuzzy output of the inference engine is mapped to a crisp value that provides the most accurate representation of the fuzzy set. There are numerous methods for defuzzification, among which centroid/center of gravity (COG) method, bisector of area (BOA) method and mean of maxima (MOM) method. The centroid method provides a crisp value based on the center of gravity of the fuzzy set, thus, for a discrete membership function,

$$COG = \frac{\sum_{i=1}^{n} x_i \cdot \mu_A(x_i)}{\sum_{i=1}^{n} \mu_A(x_i)}$$

While for a continuous membership function the expression is

$$COG = \frac{\int_a^b \mu_A(x) \cdot x \, dx}{\int_a^b \mu_A(x) \, dx}$$

where μ_A represents the membership of the fuzzy sets and the value of the membership is represented as x_i [15] [21].

For its part, the bisector method consists in the calculation of the position under the curve where the area on both sides are equal. This method is defined as

$$\int_{\alpha}^{BOA} \mu_A(x) \, dx = \int_{BOA}^{\beta} \mu_A(x) \, dx$$

where μ_A represents the membership of the fuzzy sets, $\alpha = \min\{x \mid x \in X\}$ and $\beta = \max\{x \mid x \in X\}$, X denoting the universe of discourse [12] [17].

In the mean of maxima method, the defuzzified value is taken as the element with the highest membership values; in the case where more than one element has maximum membership values, it is considered the mean value of the maxima. The defuzzified value is given by

$$MOM = \frac{\sum_{x_i \in M} x_i}{|M|}$$

here $M = \{x_i \mid \mu_A(x_i)$ is equal to the height of the fuzzy set A$\}$ and $|M|$ is the cardinality of the set M [17].

B. Latest relevant applications of FIS

Fuzzy logic is extensively used for practical and commercial purposes given its acceptable reasoning. Furthermore, it has the ability to deal with uncertainty in a wide range of fields. Some examples of the latest relevant applications include medicine, transportation systems, security, and finance; FIS support these areas in the process of decision-making and controlling certain aspects, e.g. in the specific case of medicine, are helpful in diagnosis and in transportation systems control train schedules [5].

C. Limitations/downfalls of using FIS

Despite the advantages a FIS has, it also presents some limitations and downfalls. Since these inference systems mostly work on inaccurate data and inputs, their accuracy may be compromised, besides they are completely dependent on human knowledge and expertise, requiring regular updates of inference rules, another drawback of using a FIS is the need to thoroughly test for validation and verification. [5]

III. Problem's Definition

In the last 10 years, cloud services have taken over a large part of the tech industry and cloud transitions have become common due to the contributions it gives to businesses' effectiveness and long-term cost savings. [6] Thence, aspects regarding security and privacy top the list of priorities of cloud service users considerations being a great cause of

apprehension when opting for these services.

In order to perform an analysis of the security of cloud service providers, an evaluation of a cloud service provider from a user is collected. This evaluation lays on various factors, namely, compliance, access control, auditability, and encryption. The cloud compliance aspect refers to the conformity with regulatory standards of cloud usage in accordance with industry guidelines and local, national, and international laws [10]; access control concerns to the ability to restrict access to information stored on the cloud, which allows companies to ensure their information is secured and helps minimize risk [9]. As in any business, auditability is crucial to check and improve data availability and make considerations on the overall performance and security aspects that should be ensured by the cloud service provider [1]. The last criterion considered to evaluate cloud computing security is encryption, which consists of the transformation of data from its original plain text format to an unreadable format before it is transferred and stored in the cloud [11].

Applying a fuzzy logic approach to model this problem, the evaluation factors referenced above are established as the fuzzy variables. Each can assume a value multiple of 5, ranging from 0 to 100, i.e., the universe of discourse is given as

$$U = \{\, 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, \\ 55, 60, 65, 70, 75, 80, 85, 90, 95, 100 \,\}$$

To allow the decision-making process possible there is the necessity of formulating inference rules. This rule base consists of a set of IF-THEN rules, where the antecedents are the evaluation factors, compliance, access control, auditability and encryption; and the consequent is the security provided. Each evaluation factor is divided into levels according to its performance; this description is specified in Table I.

Linguistic Variables	Membership Degree	Description
Harmful	$[0, 20]$	Presents danger
Risky	$[10, 40]$	Potentially harmful
Average	$[30, 70]$	Uncertain about security
Great	$[60, 90]$	Better than most
Exceptional	$[80, 100]$	Near flawless

TABLE I
DEFINITION OF LINGUISTIC VARIABLES.

Since the evaluation factors depend on each other, the inference rules are constructed based on the "and" operator. Here are presented the inference rules formulated to enable the evaluation of service providers' security using the linguistic form in this particular case of study

$$\text{If } (CP = H) \wedge (AC = R) \wedge (AU = H) \wedge (EC = G), \\ \text{then } (SC = R). \tag{1}$$

$$\text{If } (CP = R) \wedge (AC = AV) \wedge (AU = H) \wedge (EC = G), \\ \text{then } (SC = AV). \tag{2}$$

$$\text{If } (CP = EX) \wedge (AC = G) \wedge (AU = G) \wedge (EC = G), \\ \text{then } (SC = G). \tag{3}$$

$$\text{If } (CP = EX) \wedge (AC = EX) \wedge (AU = G) \wedge (EC = G), \\ \text{then } (SC = G). \tag{4}$$

where CP, AC, AU, EC, and SC stand, respectively, for compliance, access control, auditability, encryption and security.

IV. RESEARCH/WORK QUESTION

As stated before, this study aims to analyze the security of cloud service providers through a FIS. Taking this into consideration, a code was developed in *Python* on which it is given a brief description.

The libraries used in the code were *numpy*, *skfuzzy* and *matplotlib*. *Numpy* offers comprehensive mathematical functions, random number generators, linear algebra routines, array handling and more. Through the usage of *skfuzzy* is possible to implement many tools that are useful for fuzzy logic. To help visualize the results we use the *matplotlib* library.

In this project, we perform two FIS and apply two techniques to obtain the output, i.e, the security index of a cloud service provider. The first technique is applied with the same inputs as the ones considered in the first case of the paper [19] and then with inputs of another user, values that are assumed by us. Therefore, it is possible to confirm the security index of a cloud service provider obtained in the article and also explore an optimal scenario where the security index would be considered as great, according to the definition of linguistic variables; which can serve as a self-evaluation for cloud service providers of the type of evaluation they should expect from their users. We then applied the second technique for these two scenarios as well. However, using this technique we will still explore the Łukasiewicz and Gödel operators, as well as different defuzzification methods besides the centroid, such as the bisector method or the mean of maximum method. Finally, we make a brief comparison of these two techniques/algorithms used.

A. Technique I - With inputs from original paper [19]

In order to analyze the article's first case of study, we assign the value given by a cloud service user to the respective evaluation factors, which are presented in the following table Table II.

Evaluation Factors	Input Values
Compliance	15
Access Control	35
Auditability	10
Encryption	75

TABLE II
ANTECEDENTS AND INPUT VALUES.

Taking into account the input values, the two rules which fulfill the conditions stated in Table II, are rule 1 and rule 2 described in section Problem's Definition.

Collecting the rules we get the results from the fuzzy Control System by applying the *Python* functions `ctrl.ControlSystem()` and `ctrl.ControlSystemSimulation()`. Its application results in the process of defuzzification using the centroid method, through which we obtain its value that represents the security index.

B. Technique I - With hypothetical inputs

In this section, the approach is the same as in Technique I, however, the inputs considered are from a hypothetical cloud service user, shown in Table III.

Evaluation Factors	Input Values
Compliance	91
Access Control	82
Auditability	73
Encryption	75

TABLE III
ANTECEDENTS AND HYPOTHETICAL INPUT VALUES.

The inference rules to apply in this case are the rule 3 and rule 4 described in Problem's Definition.

C. Technique II - With inputs from original paper [19]

Using the previous method (Technique I), we applied a function that does the FIS directly, that is, we don't have an explicit line/set of lines of code for the application of T-norms or S-norms (T-co-norms), for example, or the strength of each rule, the exact membership values for a given input, or even the process of defuzzification and application of the centroid method. These steps are not explicit because they are inside the function `ctrl.ControlSystemSimulation()`.

This way, we are also going to apply a method that computes everything by hand, step by step. We start by, for each antecedent, as well as for the consequent, creating the fuzzy membership functions by using `fuzz.trimf()` which is a triangular membership function generator, used to define the membership functions according to the levels of linguistic variables previously described in Table I.

We then proceed to visualize the membership functions, for each antecedent, as well as the consequent, using the *Python* function `matplotlib.pyplot.plot()`. To get the exact value of the membership degree we use the function `fuzz.interp_membership()`, for an antecedent and its corresponding input value.

Once this is done, we move on to the fuzzy inference rules, which, as we have already seen, are the rules in (1) and (2). Now, in these rules we only have the "and" condition, therefore we must apply the T-norm operator defined as $T(a, b) = \min(a, b)$, using the function `np.fmin()`. From this, we can then extract the strength of each rule. Then, we aggregate the two rules together by using the function `np.fmax()`.

To complete all stages of a FIS, it is necessary to convert the fuzzy set output into crisp values, hence we apply defuzzification explicitly, using the function `fuzz.defuzz()` with the aggregated rules and the centroid method. With the `.plot()` function is possible to visually represent the defuzzification and the fuzzy membership for the security output, as well as the centroid and its value.

In addition to the centroid method, we also explore other methods, such as the bisector method and the mean of maximum method, to better understand the possible variations of this process. These three methods were selected considering their suitability to the problem approached. Methods such as min of maximum or max of maximum would not make sense applied to this problem.

D. Technique II - Łukasiewicz's Operators

Since the goal of this study is to analyze the security readiness of cloud service providers and for that purpose is necessary to consider evaluation factors that are dependent on each other, we opted for the exploration of T-norms, more specifically, for Łukasiewicz's operators in this section.

Therefore, taking the rules of (1) and (2) again and applying the Łukasiewicz's operator for the T-norm, we infer the following

$$\begin{aligned} T(T(T(a,b),c),d) &= \max(0, T(T(a,b),c) + d - 1) \\ &= \max(0, \max(0, T(a,b) + c - 1) + d - 1) \\ &= \max(0, \max(0, \max(0, a + b - 1) + c - 1) + d - 1) \end{aligned} \tag{5}$$

Hence, for the first rule

$$\begin{aligned} \max(0, \max(0, \max(0, (CP = H) + (AC = R) - 1) \\ +(AU = H) - 1) + (EC = G) - 1) \end{aligned} \tag{6}$$

And, for the second rule

$$\begin{aligned} \max(0, \max(0, \max(0, (CP = R) + (AC = AV) - 1) \\ +(AU = H) - 1) + (EC = G) - 1) \end{aligned} \tag{7}$$

Finally, we again apply the operator, to (1) with $(SC = R)$ and (2) with $(SC = AV)$, as stated in (1) and (2). As before, we perform the rules aggregation and defuzzification using the centroid method.

E. Technique II - Gödel's Operators

In this section, a T-norm is applied based on Gödel operator. Again, taking the rules of (1) and (2) and applying the Gödel's operator for the T-norm, we will infer the following

$$\begin{aligned} T(T(T(a,b),c),d) &= T(T(ab,c),d) \\ &= T(abc,d) = abcd \end{aligned} \tag{8}$$

Then, for the first rule

$$(CP = H) \cdot (AC = R) \cdot (AU = H) \cdot (EC = G) \tag{9}$$

And, for the second rule

$$(CP = R) \cdot (AC = AV) \cdot (AU = H) \cdot (EC = G) \tag{10}$$

We again apply the operator, to (9) with $(SC = R)$ and (10) with $(SC = AV)$ as stated in (1) and (2), respectively. We perform the aggregation of the rules, as well as the defuzzification with the centroid method and its visualization.

F. Technique II - With hypothetical inputs

As we did in Technique I, we will consider inputs from a hypothetical cloud service user, represented in Table III . This way, as previously stated the inference rules to applied in this scenario are the rule 3 and rule 4 described in Problem's Definition.

V. Results

A. Technique I - With inputs from original paper [19]

The membership functions, for each of the antecedents, are defined all as triangular. Thus, for the antecedent CP, we obtained the graph represented in Fig.1.

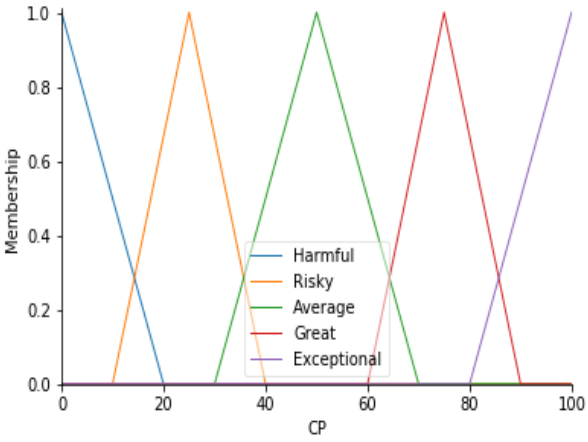

Fig. 1. CP Membership Function.

For the remaining antecedents and for the consequent, we have exactly the same behavior, represented in the Apendix, in Fig.6, Fig.7, Fig.8, Fig.9.

Performing the process of rule aggregation and defuzzification using the centroid method, results in a centroid approximately equal to 39.7, concluding that, as this value concerns the security index, this cloud service provider has average security. Graphically,

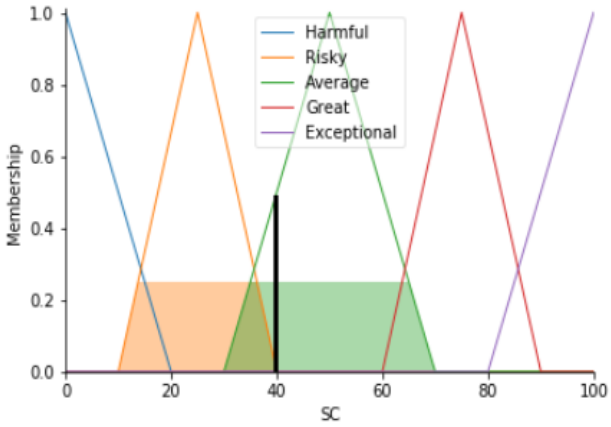

Fig. 2. Defuzzification - Technique I with inputs from the original paper.

B. Technique I - With hypothetical inputs

From the security index value, the cloud service user can evaluate whether the security that the cloud service provider offers is sufficient. In the previous section, we got a value of 39.7, approximately. Since this is not an excellent value, the user may be interested in changing cloud service provider, in order to obtain better security for himself. Therefore, we decided to develop exactly the same algorithm as before, but changing the inputs, using the ones in Table III. Computing the defuzzification by the centroid method yields the following output:

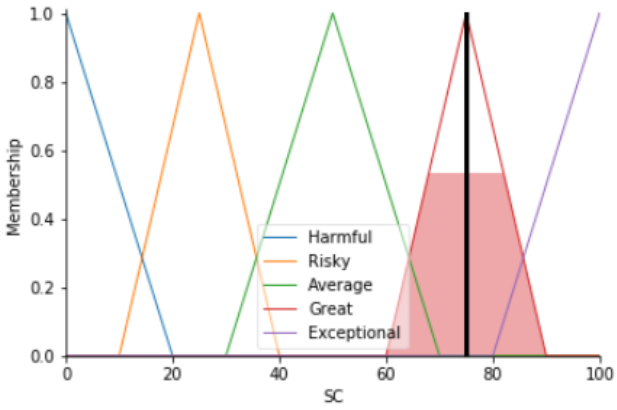

Fig. 3. Defuzzification - Technique I with hypothetical inputs.

The centroid value is, approximately, 75.0, which indicates that this is a great security index and can be interpreted as a good example for cloud service providers, which will help

them perceive what is expected of them by cloud service users, although it is not considered an exceptional model according to the definition of linguistic variables for this problem.

C. Technique II - With inputs from original paper [19]

In this section, the membership functions considered are the same as in Technique I, so we get as output Fig.10 represented in the Apendix.

To the specific inputs analyzed ($CP = 15, AC = 35, AU = 10$ and $EC = 75$) the membership degree is computed; the values are presented in the following table.

Membership Degree	H	R	AV	G	EX
CP	0.25	0.3(3)	0.0	0.0	0.0
AC	0.0	0.3(3)	0.25	0.0	0.0
AU	0.50	0.0	0.0	0.0	0.0
EC	0.0	0.0	0.0	1.0	0.0

TABLE IV
MEMBERSHIP DEGREE OF THE ANTECEDENTS.

The inference rules applied are the ones from equations (1) and (2), evaluating the strength of each rule we conclude that they both have a value of 0.25.

Through the process of defuzzification using the centroid method, we obtain:

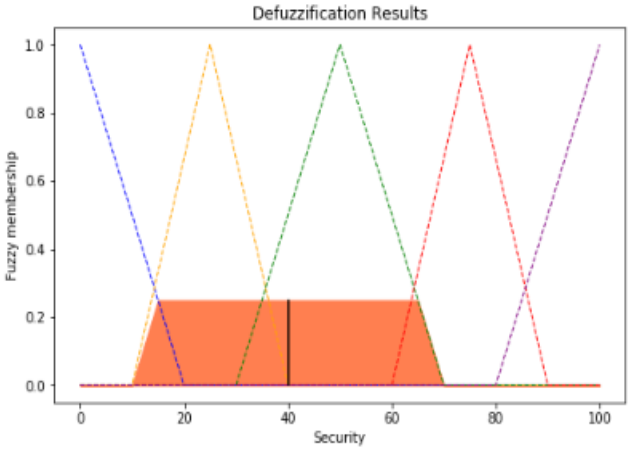

Fig. 4. Defuzzification by the centroid method - Technique II with inputs from the original paper.

Where the exact centroid value is equal to 40.0, corresponding to an average security index for the cloud service provider.

For the bisector and mean of maxima methods we got exactly the same visualization as in Fig.4 and exactly the same result.

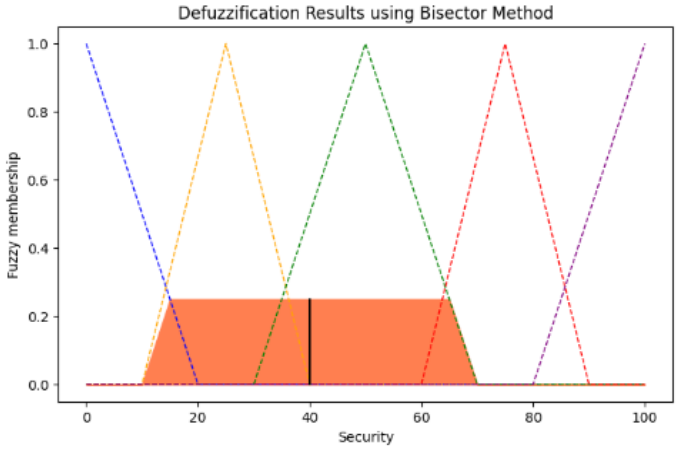

Fig. 5. Defuzzification by the bisector method - Technique II with inputs from the original paper.

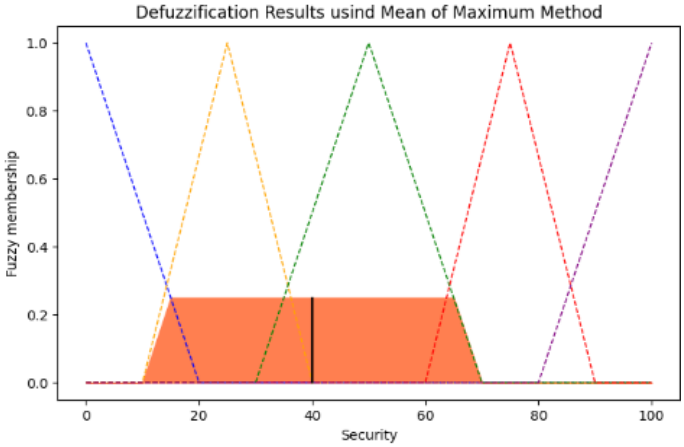

Fig. 6. Defuzzification by the mean of maximum method - Technique II with inputs from the original paper.

D. Technique II - Łukasiewicz's Operators

Applying Łukasiewicz's operator for the T-norms to the rules and calculating the weight of both, we get 0 for both. This is because when we apply this operator the maximum of the conditions is always 0. Therefore, it is not possible for us to use this operator since we would get a null area in the defuzzification step.

E. Technique II - Gödel's Operators

Applying Gödel's operators for the T-norms to the rules and computing their weight, we get approximately 0.04 for each one, a value very close to 0, a result similar to the rule strength obtained using Łukasiewicz's operator. After the rules aggregation and defuzzification by the centroid method, we obtain

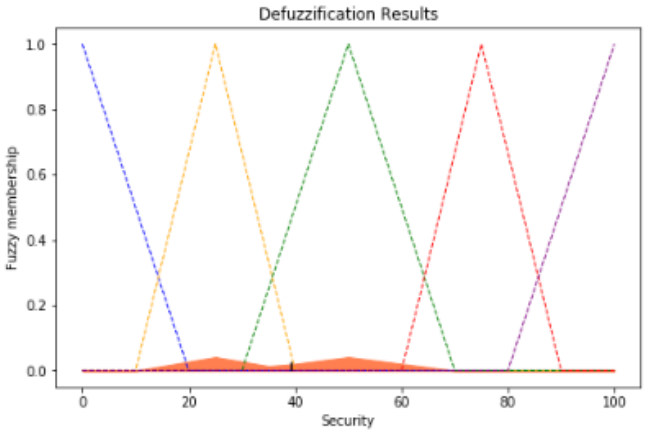

Fig. 7. Defuzzification - Technique II with inputs from the original paper.

Where we conclude that, in this case, the security index of the cloud service provider is 39.4.

F. Technique II - With hypothetical inputs

As in Technique I, here are considered the inputs of Table III, being the membership functions equal to the ones of Fig.10. Through defuzzification, using the centroid method, we get

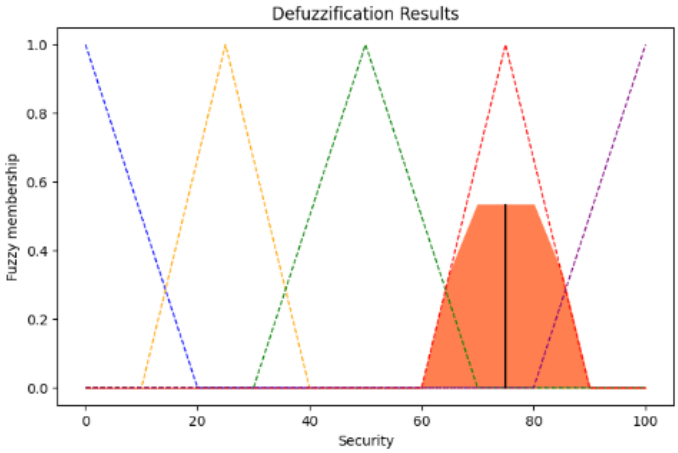

Fig. 8. Defuzzification - Technique II with hypothetical inputs.

The centroid value is equal to 75.0, as before, which indicates a great security index, as we verified in Technique I.

G. Comparison of Techniques

In this study we approached the problem of evaluating the security readiness of cloud service providers through fuzzy inference systems; throughout this process, two different techniques were used.

In Technique I, the new API for fuzzy systems was applied. As already seen, the application of T-norms or S-norms, the computation of rules strength, the exact membership degrees for a given input and the process of defuzzification is not explicit. Furthermore, we can conclude that during the defuzzification process the membership function is clipped [3].

To have a more clear understanding of the process steps, we decided to address another technique, which we named Technique II, where we compute everything step by step. Thereby, it is possible to have an insight into what each parameter does, as well as change them and distinguish each step mentioned in the methodology. Besides, this method presents defuzzified output as a scaled membership function [7].

Clipping is the most common method of correlating the rule consequent with the truth value of the rule antecedent. Here, the consequent membership function is cut at the level of the antecedent truth. However, despite the fact clipping is the most frequently used method, scaling offers a better approach for preserving the original shape of the fuzzy set, it generally loses less information. The original membership function of the rule consequent is adjusted by multiplying all its membership degrees by the truth value of the rule antecedent [13].

VI. Related Work

Recently, several models in the trust management domain have been developed, due to the inherent advantages of cloud computing. This section is dedicated to the analysis and comparison with some already created models similar to ours.

Within this theme, there are various models proposed in the literature. In the work published by Rathi and Kolekar [18], the parameters they focused on to evaluate the security of a cloud service provider were the safety of the data, individuality management, authorization, authentication, and virtualization. The input values, as in our case, consisted of the evaluation of a cloud service user on these specific parameters. The trust value is calculated by a point-based system that takes the ratio of user-gained points to the total points possible. The problem with this approach is that the user may be choosing a cloud service provider that is trustworthy in one specific parameter and not so efficient in others.

Riv et al. [20] used third-party auditors in their work to establish trust between cloud service users and cloud service providers. Despite the benefits of a third-party assessment, they were unable to use this to compute a security index for each cloud service provider.

Lastly, Kurdi et al. [14] introduced a subjective logic-based algorithm (*InterTrust*) in their work. Subjective logic is explained in the literature as a proponent of trust that has an input of uncertainty and incomplete knowledge. The main goal of this work is to prove the superiority of *InterTrust* as the only trust management system, in other words, it is not focused on the security of cloud services. However, certain parts of this research are still within our fuzzy inference

system as a trust model, such as subjective trust.

In this study, the work of Mamdani for an inference system [22] was used to define our inference rules. We also used the work of Kurdi et al. [14] to identify our subjective and objective trust definitions. What differentiates this work from the previously mentioned ones is that it takes into account the cloud service users' inputs considering their definition of trust, synthesizing the information computationally, removing the security index of the cloud service provider in question [19].

VII. Conclusion

Fuzzy inference systems have a wide range of applications, including security area. In this project, we developed fuzzy inference systems to yield a security index for cloud service providers. The fuzzy inference systems were constructed based on two techniques, in the first one, the inputs considered are described in the article on which this study was based, resulting in a security index of, approximately, 39.7. In the second technique, the security index obtained was slightly higher rounding 40.0, being this difference mainly explained by the way the defuzzied output is generated.

As stated before, the security index is considered average, which represents uncertainty about security. Therefore, users may want to look for other cloud service providers that better meet their necessities. Thus, we analyzed a hypothetical case in which the security index would be considered an optimal scenario; using both techniques we obtained a security index of 75, approximately.

Furthermore, when applying the second technique we explored, besides the centroid method, the bisector method, and the mean of maxima method, yielding the same result in all cases. In addition, we experimented different operators for t-norms, namely, Łukasiewicz's operators and Gödel's operators. From Łukasiewicz's operators, we could not obtain any defuzzification area, since the rules applied have both rule strengths equal to zero. For its part, Gödel's operators showed a defuzzification area very close to zero, nonetheless, it was possible to calculate the center of gravity of rules, concluding that the security index, in this case, was equal to 39.4.

Despite the different results among the used methods, they range from 39.4 and 40, from which we infer the consistency of results.

References

[1] Cloud audit in a nutshell. https://www.netguru.com/blog/cloud-audit-guide. Accessed: 2022-12-14.

[2] Cloud service provider. https://www.techtarget.com/searchitchannel/definition/cloud-service-provider-cloud-provider. Accessed: 2022-12-14.

[3] Fuzzy control systems: The tipping problem. https://pythonhosted.org/scikit-fuzzy/auto_examples/plot_tipping_problem_newapi.html. Accessed: 2022-12-12.

[4] Fuzzy logic: Definition, meaning, examples, and history. https://www.investopedia.com/terms/f/fuzzy-logic.asp. Accessed: 2022-12-15.

[5] Fuzzy logic in artificial intelligence: Architecture, applications, advantages disadvantages. https://www.upgrad.com/blog/fuzzy-login-in-artificial-intelligence/. Accessed: 2022-12-15.

[6] How the cloud has evolved over the past 10 years. https://www.dataversity.net/how-the-cloud-has-evolved-over-the-past-10-years/. Accessed: 2022-12-15.

[7] The tipping problem - the hard way. https://pythonhosted.org/scikit-fuzzy/auto_examples/plot_tipping_problem.html. Accessed: 2022-12-12.

[8] What is a cloud service provider? https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-a-cloud-provider/. Accessed: 2022-12-14.

[9] What is access control in cloud computing. https://www.baass.com/faq/what-is-access-control-in-cloud-computing. Accessed: 2022-12-14.

[10] What is cloud compliance. https://www.trendmicro.com/en_nz/what-is/cloud-security/cloud-compliance.html. Accessed: 2022-12-14.

[11] What is cloud encryption. https://www.crowdstrike.com/cybersecurity-101/cloud-security/cloud-encryption/. Accessed: 2022-12-14.

[12] M. V. C. Rao Aarthi Chandramohan and M. Senthil Arumugam. Two new and useful defuzzification methods based on root mean square value. *Soft Computing*, 2006.

[13] Elmer P. Dadios. *Fuzzy Logic - Controls, Concepts, Theories and Applications*. 2012.

[14] H. Kurdi, A. Alfaries, and Al-Anazi A. Alkharji S. Addegaither M. Altoaimy L. AhmedSH. A lightweight trust management algorithm based on subjectivelogic for interconnected cloud computing environments. *J Supercomput75(7):3534–3554*, 2019.

[15] N. Mogharreban and L. F. DiLalla. Comparison of Defuzzification Techniques for Analysis of Noninterval Data. *IEEE*, 2006.

[16] Vilem Novak, Irina Perfiljeva, and Jiri Mockor. *Mathematical principles of fuzzy logic*. Kluwer Academic Publisher, Boston, 1999.

[17] D. K. Pratihar. Soft Computing. *Narosa Publication*.

[18] S. R. Rathi and V. K. Kolekar. Trust Model for Computing Security of Cloud. *Fourth International Conference on Computing Communication Controland Automation (ICCUBEA)*, 2018.

[19] S. Rizvi, J. Mitchell, and A. et al. Razaque. A fuzzy inference system (FIS) to evaluate the security readiness of cloud service providers. *J Cloud Comp 9, 42*, 2020.

[20] J. Kissell S. Rizvi, J. Ryoo and W. Aiken. A Stakeholder Oriented AssessmentIndex for Cloud Security Auditing. *The 9th ACM International Conference onUbiquitous Information Management and Communication*.

[21] Jean J. Saade and Hassan B. Diab. Defuzzification Methods and New Techniques for FuzzyControllers. *Iranian Journal of Electrical and Computer Engineering*, 2004.

[22] F. Topaloglu and H. Pehlivan. Comparison of Mamdani type and Sugenotype fuzzy inference systems in wind power plant installations. *6th International Symposium on Digital Forensic and Security (ISDFS)*, 2018.

APPENDIX

A. Membership Functions - Technique I with paper inputs

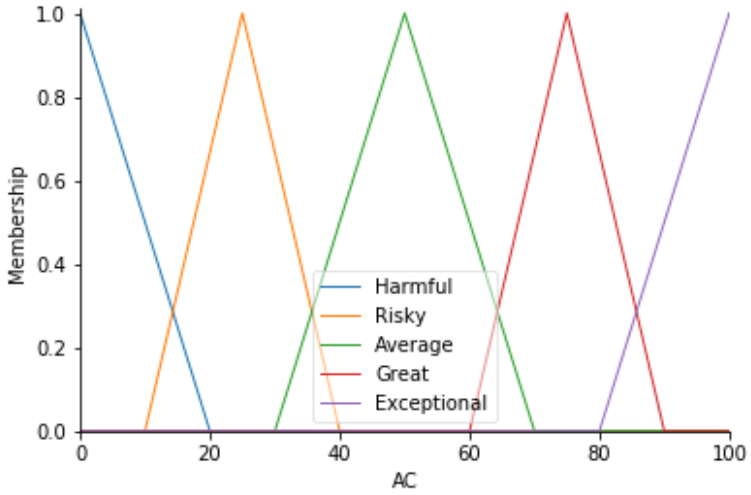

Fig. 9. AC Membership Function.

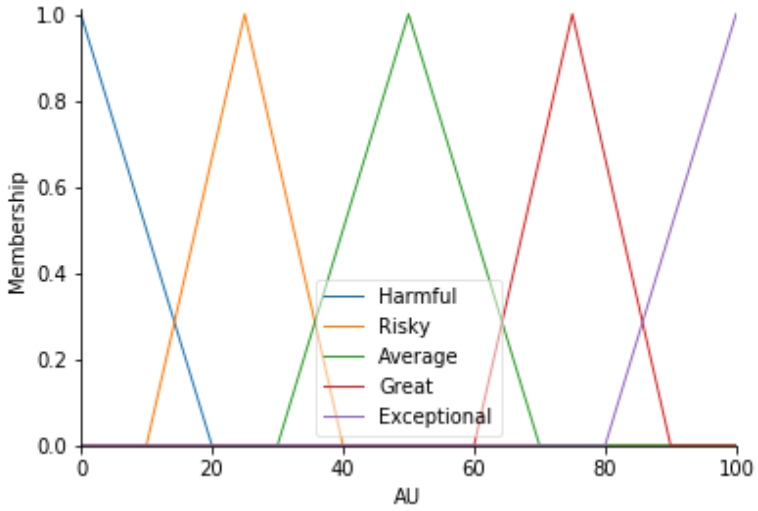

Fig. 10. AU Membership Function.

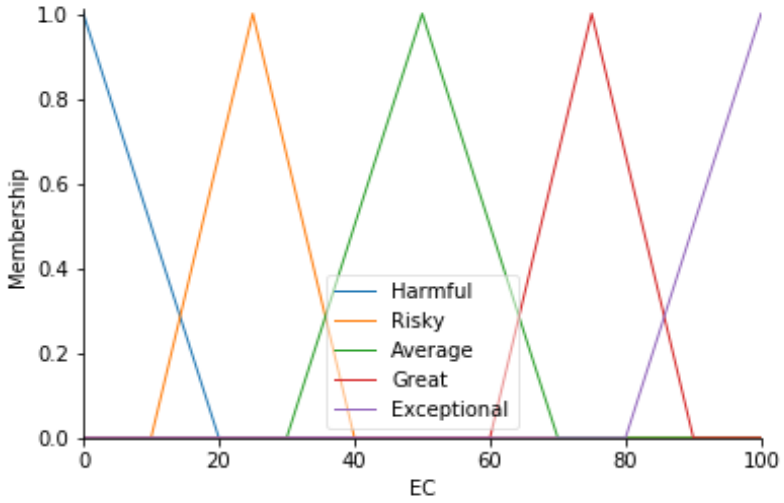

Fig. 11. EC Membership Function.

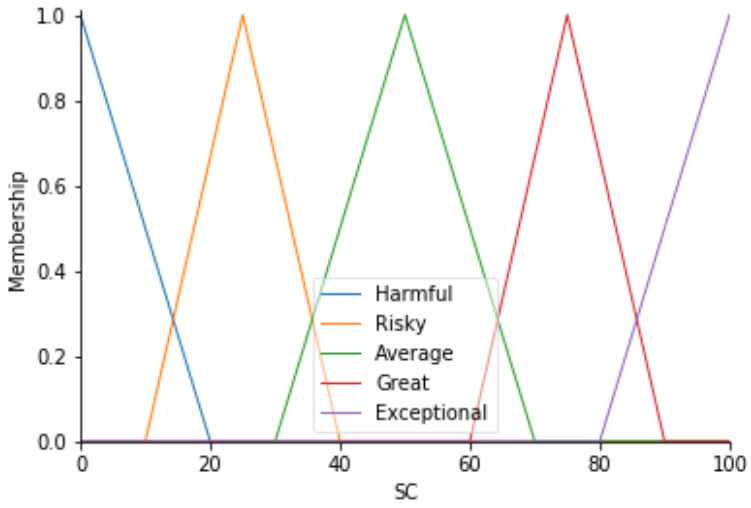

Fig. 12. SC Membership Function.

B. Membership Functions - Technique II with paper inputs

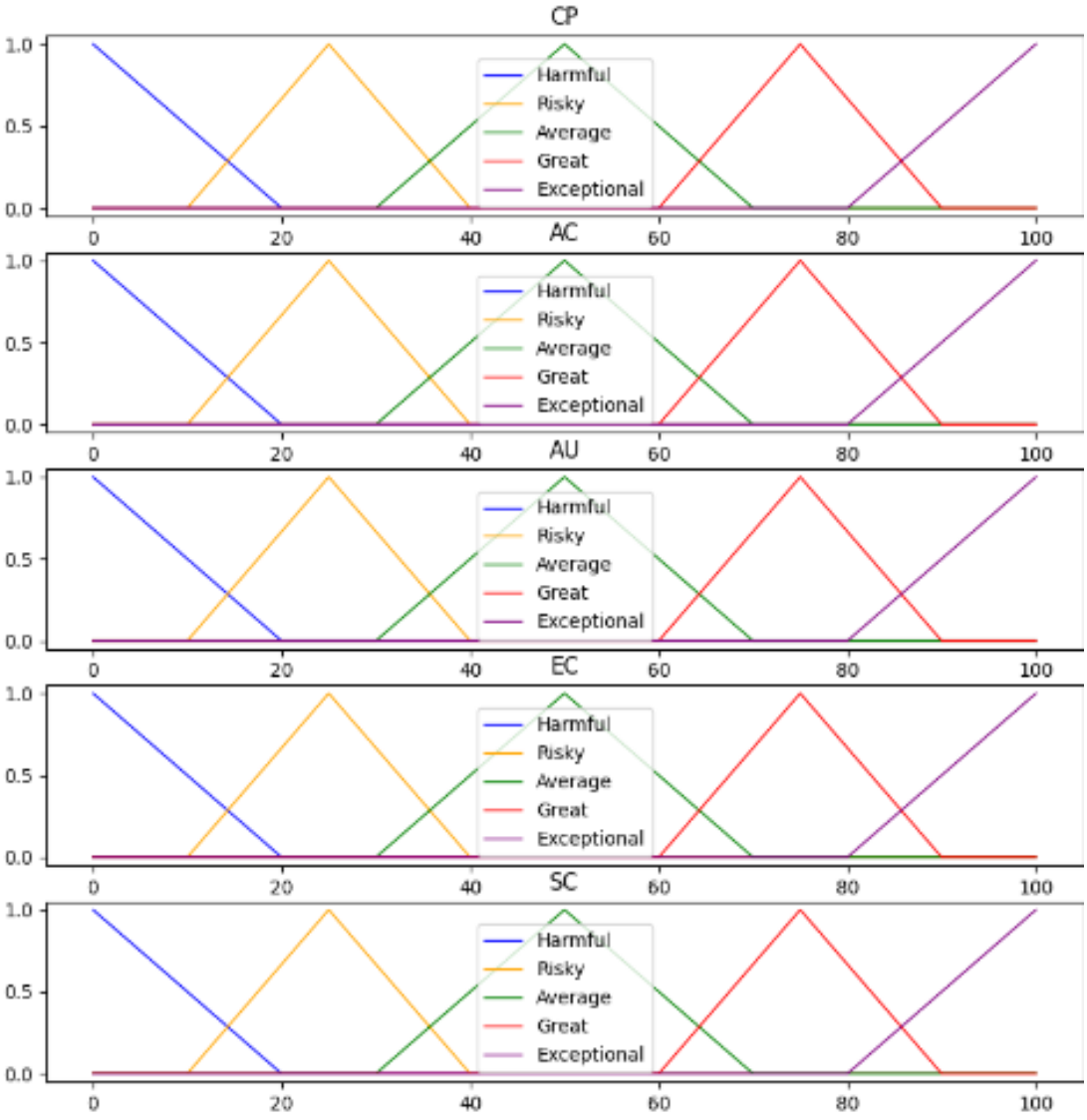

Fig. 13. Membership Function of CP, AC, AU, EC and SC.