


Article

Web User Trust Evaluation: A Novel Approach Using Fuzzy Petri Net and Behavior Analysis

Zenan Wu ¹ , Liqin Tian ^{1,2,*}, Yi Zhang ² and Zhigang Wang ¹

¹ Department of Computer, Qinghai Normal University, Xining 810008, China; chris_wu11.1@stu.qhnu.edu.cn (Z.W.); 201933341004@stu.qhnu.edu.cn (Z.W.)

² Department of Computer, North China Institute of Science and Technology, Beijing 101601, China; 201908522456zy@ncist.edu.cn

* Correspondence: tianliqin@tsinghua.org.cn; Tel.: +86-185-1146-5255

Abstract: With the development of society and information technology, people's dependence on the Internet has gradually increased, including online shopping, downloading files, reading books, and online banking. However, how to ensure the safety and legitimacy of these network user behaviors has become the focus of attention. As we all know, cybersecurity and system resilience originate from symmetry. Due to the diversity and unpredictability of cyber-attacks, absolute cybersecurity is difficult to achieve; system resilience indicates that protecting system security should shift from resisting attacks to ensuring system continuity. The trust evaluation of network users is a research hotspot in improving network system security. Aiming at the defects of incomplete evaluation processes and inaccurate evaluation results in current online user behavior trust evaluation methods, this paper combines the basic principles of online user trust evaluation and proposes a trust evaluation model that combines fuzzy Petri nets with user behavior analysis. First, for "unfamiliar" users, we used fuzzy Petri nets to calculate the user's recommended trust value as the system's indirect trust value; next, we used the user's behavior record as evidence to conduct direct trust evaluation on the user to obtain the system's direct trust in the user's value; finally, the two calculation results were combined to obtain the user's comprehensive trust value. In terms of experimental verification, the experimental data came from a self-developed e-book management system. Through theoretical analysis and simulation results, it was shown that the model met the optimization conditions of subjective and objective relative balance, the evaluation process was more complete, and the trust evaluation values of network users could be obtained more accurately. This evaluation method provides solid theory and research ideas for user credibility judgment of key network basic application platforms such as online shopping malls, online transactions, and online banking.



Citation: Wu, Z.; Tian, L.; Zhang, Y.; Wang, Z. Web User Trust Evaluation: A Novel Approach Using Fuzzy Petri Net and Behavior Analysis. *Symmetry* **2021**, *13*, 1487. <https://doi.org/10.3390/sym13081487>

Academic Editors: José Carlos R. Alcántud and Basil Papadopoulos

Received: 9 July 2021

Accepted: 10 August 2021

Published: 13 August 2021

Keywords: network security; user behavior; trust evaluation; fuzzy Petri net; weight optimization

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, people have used the Internet to share information and participate in various network activities such as downloading documents, shopping online, watching videos, playing games, etc. However, these all depend on the mutual trust of each participant in the network environment. Although there are many technologies to ensure the normal operation of the network system, most of them are used to strengthen security [1]. User verification only relies on traditional identity authentication, which is obviously a defect because a user who accesses the network can easily enter or leave the network. Therefore, they are likely to damage the network system resources in order to further their own interests such as impersonating someone else's identity. Information is used to steal confidential data within the company; even personnel within the company use the convenience of their own positions to steal confidential data. The occurrence of these two behaviors can evade the checkpoint of identity authentication and cause the loss of network system resources [2].

Therefore, there is an urgent need to find an effective method that can protect network security and restrain the behavior of malicious users to ensure the security of data resources in the network system [3,4].

2. Related Work on Web User Trust Evaluation

The problem of user trust evaluation in the network environment has always been the focus of researchers. The introduction of trust attributes into the access control model can improve the shortcomings of the existing model, thereby enhancing the security of the network system [5]. In recent years, many scholars have conducted more in-depth research and achieved many meaningful results. Jiang et al. [6] proposed the use of credibility graphs to calculate user trust values, comparatively reviewed two categories of graph-simplification-based and graph-analogy-based approaches and discussed their individual problems and challenges. As far as we know, the disadvantage of this method is that it may produce results that exceed the trust range. In addition, the algorithm has high time and space complexity and is not suitable for large-scale networks. Xiao et al. [7], in order to solve the problem of user choice of high-trust service providers in online social networks, proposed a scheme called RHT (recommendation from high trust value entities) to evaluate the trust degree of the service recommended. When calculating the comprehensive trust value in this article, the authors did not fully discuss the weight coefficients for measuring the direct trust value and the indirect trust value. Therefore, the accuracy of the evaluation results needs to be improved. Wang et al. [8] used machine learning algorithms to calculate a user's trust value, the user's privacy leakage risk was evaluated through information flow prediction, and the user's privacy leakage risk was mapped to the trust evidence and combined through the improved evidence combination rules of evidence theory. The shortcoming of this study was that there was no in-depth study on the calculation of the weight of evidence. Jiang et al. [9], in order to solve the problem of the spread of trust value in online social networks, proposed a modified flow-based trust evaluation scheme, GFTrust, in which they addressed path dependence using network flow and modeled trust decay with the leakage associated with each node. The literature's hypothesis on the initial trust value is questionable. Yang et al. [10], taking into account the high degrees of uncertainty, complexity, and dynamics of user behavior in OSN, introduced the cloud model theory into user behavior trust evaluation, and a user behavior evaluation scheme combined with entropy weight was proposed. This method is an improvement of the entropy method. Calculating the weight of each attribute overcomes the limitations of subjective weight distribution to a certain extent. The limitation of this method is that there is no difference between subjective weight and objective weight in the evaluation results. Therefore, the accuracy of the evaluation model needs to be improved. Gong et al. [11], using the influence of transaction attributes and social relationships on user trust, proposed a comprehensive trust model. This model improves the granularity of trust evaluation and improves the discrimination of recommended information, to a certain extent. However, when calculating the comprehensive trust value, the author did not discuss in detail the influence of weights on the evaluation results. The importance of transaction attributes and social relationships to the results is completely determined by the user's personality. This approach is questionable and requires more in-depth research. Ceolin et al. [12], modeling trust relying on user reputation, user demographics, and from provenance, demonstrated that using provenance and demographic information was beneficial for the accuracy of trust assessments. This method also does not give a clear solution to the distribution of evidence weight. At the same time, this method ignores the importance of recommendation trust. Ghosh et al. [13] were dedicated to the study of stock price forecasting, treating stock price forecasting as a binary classification problem and applying kernel principal component analysis (KPCA) to feature extraction of technical indicators. User behavior trust evaluation can also be regarded as a binary classification problem in certain application situations, namely, trusted users and untrusted users. In addition, feature extraction is an important process in the classification problem. Liu et al. [14],

in order to let cloud users find cloud services which satisfied performance preferences, used the comprehensive trust cloud center of gravity assessment method (CCGE) to calculate the trust level of cloud services, introduce the membership theory into the trust evaluation model, and then establish a precise trust relationship between cloud users and cloud services based on user performance requirements. This method does not include a detailed discussion on the weight distribution of direct trust value and indirect trust value. In addition, the division of trust levels is not convincing. Wang et al. [15] pointed out the current problems facing trust evaluation such as the lack of necessary evaluation data, the need for big data processing, etc., and conducted a comprehensive investigation of trust evaluation based on machine learning. Although machine learning technology has been applied in many fields, it is still in its infancy in terms of network user trust evaluation. The current shortcomings of machine learning used in trust evaluation mainly include: high requirements for data quality, strong objectivity of trust evaluation results, lack of subjective and objective combination, and prone to “overfitting” phenomenon, etc. Zhou et al. [16] proposed a dynamic trust evaluation model based on affective intensity computing, which used fuzzy logic operators to calculate partial trust, feedback trust, and overall trust. The advantage of this model lies in the introduction of a feedback trust mechanism, but the disadvantage is that it does not solve the influence of weights on the evaluation results. Li et al. [17] proposed a trust model based on fuzzy similarity. They used fuzzy similarity theory to process evaluation messages and obtain the rules of node behavior by integrating evaluation information. In addition, they proposed a trust update algorithm for malicious and selfish nodes. The advantage of this method is that the Kalman principle is used to establish a trust value update mechanism, which satisfies the dynamics of the trust value. However, the shortcoming of this model lies in the lack of consideration of subjective factors.

Through the above analysis, it can be found that although researchers have used a variety of methods to calculate the trust values of users from different perspectives, they all have certain limitations. For example, some methods focus more on subjectivity while ignoring the objectivity of user behavior evidence; some combine the subjective and objective but do not consider the recommendation of the recommender, or the subjective and objective weights are not discussed in detail. In short, there is currently no complete evaluation system that can integrate these methods.

In order to solve the above problems, we propose a new trust evaluation method that combines the factors of direct trust and indirect trust. When calculating the direct trust value, the influence of subjective and objective weights on the calculation result is optimized; when calculating the indirect trust value, the advantages of the efficient modeling of Petri net theory [18] and fuzzy theory are used to make the model more suitable for handling the fuzzy trust evaluation process. Therefore, the calculation result is considered more reasonable. Theoretical analysis and simulation verify the performance improvement compared with the existing mechanism. The main contributions of this paper are summarized as follows:

- (1) We propose a user behavior trust evaluation method that integrates subjective and objective influencing factors, and this method optimizes the subjective and objective weights that affect user behavior evidence. In this way, the network user behavior trust evaluation can satisfy the relative balance of subjective and objective, and the accuracy of the evaluation results will be higher;
- (2) For unfamiliar users, we use fuzzy Petri nets to model and analyze such users and obtain their initial trust values through the recommendations of other network users. This provides a solution for solving the “cold start” problem of user trust value. Next, we use the results as the indirect evaluation value and the direct evaluation value of user behavior to optimize the configuration to obtain the comprehensive trust evaluation value of the user. The evaluation process is more complete and more reasonable.

The rest of this article is organized as follows: Section 3 introduces the basic principles

of web user trust evaluation. Section 4 introduces the detailed design of an indirect trust evaluation scheme using a fuzzy Petri net. Section 5 introduces the detailed design of the direct trust evaluation scheme using user access behavior. We integrate direct trust and indirect trust in Section 6 and evaluate the performance of the trust model through simulation experiments. Finally, conclusions and further research potential are discussed in Section 7.

3. Basic Knowledge of Web User Trust Evaluation

3.1. The Basic Principle

The WEB user trust evaluation method [19,20] draws on the evaluation of trust in social sciences, so the proposed evaluation method follows the following principles [21,22]:

- (1) In the calculation of the trust value, the importance of evidence is inversely proportional to the interaction time interval;
- (2) Trust is a long-term cumulative process;
- (3) Overall trust includes direct trust and indirect trust, with direct trust as the mainstay and indirect trust as the supplement;
- (4) The trust value should “slowly rise and quickly fall”;
- (5) The trust value is a dynamic value, which is constantly updated with time and behavior.

3.2. The Basic Definitions

Trust refers to the judgment of the relying party (usually the service provider) on the object (usually the user accessing the service) based on past experience and data.

User behavior evidence includes the behavior records generated by the user when accessing the service. These behavior records are mathematically quantified and can be used as basic data to evaluate the user’s trust value.

Direct trust is the mutual trust relationship established by two entities through mutual communication.

Indirect trust is an indirect trust relationship established through the recommendation of a third party.

Trust value refers to the degree of trust the relying party has in the object.

3.3. The Framework of Trust Evaluation

The web user trust evaluation model proposed in this paper is shown in Figure 1. When a user accesses a network system, their identity information is first verified. This process is the first verification of trust evaluation. Next, the network system uses the recommended trust value provided by other service providers of the same type to perform indirect trust evaluation on the user. This process is the second verification of the trust evaluation; only users who pass the above two verifications can access the network system normally. Next, the system directly trusts the user based on the user’s access behavior evaluation; this process is the third verification of trust evaluation. Therefore, in our proposed trust evaluation model, both indirect trust evaluation and direct trust evaluation were included. In addition, in the direct trust evaluation, we solved the problem of optimal allocation of subjective and objective weight selections.

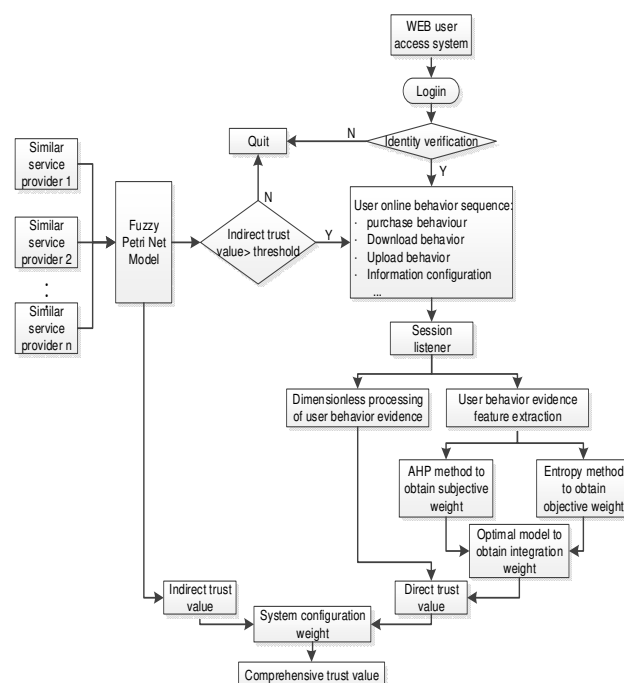


Figure 1. Web user trust evaluation flowchart.

4. Indirect Trust Evaluation Based on a Fuzzy Petri Net

4.1. Fuzzy Petri Net

A Petri net is suitable for describing asynchronous and concurrent computer system models. Its advantage is that it has strict mathematical expressions and intuitive graphical expressions. Therefore, Petri nets can also be used for network security evaluation modeling [23]. In addition, fuzzy theory is often used to resolve the uncertainty of trust relationships.

The fuzzy Petri net is an extension of Petri net [24], which combines the advantages of Petri net and fuzzy theory. Therefore, fuzzy Petri net is a powerful tool for effective modeling of fuzzy knowledge [25,26].

When a WEB user accesses the system for the first time, because there is no evidence of the user's historical behavior in the system, the system cannot perform an initial evaluation of the user. Regarding the initial trust value setting of such users, the literature [27] sets the user's initial trust value to a relatively low value. Although this setting can improve the security of the system to a certain extent, it will affect the real reliability of trust in the user's interactive experience. Therefore, this paper uses a fuzzy Petri net to calculate the user's recommended trust value and uses the result as a component of the WEB user trust evaluation, making the final trust evaluation result more reasonable.

The trust evaluation model based on fuzzy Petri nets can map general fuzzy inference rules through the representation and reasoning of fuzzy rules. The fuzzy Petri net model of trust evaluation between entities is shown in the following Figure 2:

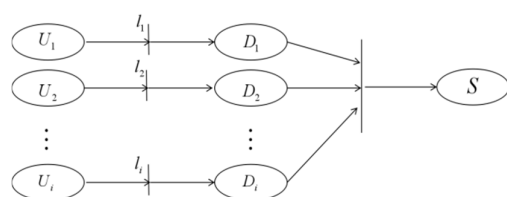


Figure 2. Fuzzy Petri net model for trust evaluation.

Among them, U_i represents the i -th recommended value of the user by the service provider of the same system, D_i represents the i -th recommended trust value of the target system accepting the same service provider, l represents the threshold of the inference rule, and S represents the comprehensive recommended trust value.

Definition 1. The user behavior trust evaluation structure of the fuzzy Petri net is defined as a 6-tuple:

$$UBTA - FPN = (P, T, I, O, \Gamma, S^0)$$

$P = \{p_1, p_2, \dots, p_m\}$ is a collection of places, where each place p_i represents a fuzzy proposition; $T = \{t_1, t_2, \dots, t_n\}$ is the set of changes, where each change t_i represents the occurrence of a fuzzy rule; $I = \{\alpha_{ij}\}$ is the input matrix representing the input relationship from p_i to t_j , where $i = 1, 2, \dots, m$; $j = 1, 2, \dots, n$ and $\alpha_{ij} = \begin{cases} (0, 1] & , \text{ if } p_i \text{ is the input place of } t_j \\ 0 & , \text{ otherwise} \end{cases}$, representing the corresponding weight on the connecting arc from p_i to t_j ; $O = \{\beta_{ij}\}$ is the output matrix representing the output relationship from t_j to p_i , and $\beta_{ij} = \begin{cases} (0, 1] & , \text{ if } p_i \text{ is the input place of } t_j \\ 0 & , \text{ otherwise} \end{cases}$, representing the corresponding weight on the connecting arc from t_j to p_i ; $\Gamma = (\tau_1, \tau_2, \dots, \tau_n)$, $\tau_j \in [0, 1]$, represents the threshold of the fuzzy rule; that is, τ_j represents the start threshold of transition t_j ; $S^0 = (s_1, s_2, \dots, s_m)^T$ is the initial marking state of the place node, or, that is, the credibility of the initial proposition where $s_i \in [0, 1]$, and $i = 1, 2, \dots, m$.

4.2. Indirect Trust Value Based on a Fuzzy Petri Net

According to the reasoning steps of the fuzzy Petri net, a comprehensive recommendation trust value can be obtained. The process is as follows:

- Step 1. Calculate the reliability of the equivalent fuzzy input: $E = I^T \cdot S^0 = [e_1, e_2, \dots, e_n]$;
- Step 2. Delete the items whose credibility is less than the threshold in the input credibility E ; if $e_j < \tau_j$, then $e_j = 0$, otherwise e_j remains unchanged, where $j = 1, 2, \dots, n$ and update credibility $E \rightarrow E'$;
- Step 3. Calculate the reliability of the equivalent modulus output $E'' = O \cdot E'$;
- Step 4. Calculate the credibility of all propositions $S^1 = S^0 \oplus E''$, where \oplus is an operation that takes the larger value between the two;
- Step 5. Iterate repeatedly. If $S^k = S^{k-1}$ appears in k iterations, the reasoning ends.

Through the above reasoning process, the user's indirect trust value $T_{indirect}$ can be obtained:

$$T_{indirect} = s_m^k \quad (1)$$

5. Direct Trust Evaluation Based on User Behavior Analysis

5.1. Obtaining Evidence of User Behavior

Evidence of user behavior usually refers to the specific manifestation of a series of operations performed by the user. Obtaining effective evidence is the prerequisite for user behavior authentication. The commonly used methods to obtain user behavior evidence are as follows:

- (1) Using an intrusion detection system such as Sguil or Tcpdump on PC [28] for network security analysis; this method can collect user real-time event activities, such as the number of user visits, etc.;
- (2) Using network traffic anomaly detection tools such as ENTVIS [29], various protocols of the gateway can be obtained, and the transmission rate of data packets can be viewed;
- (3) Analyzing user behavior through Web logs [30];
- (4) Using Ajax-based click stream capture tools such as a series of user operations on the mouse and certain operations on web pages to obtain user action information [31].

5.2. Standardized Processing of User Behavior Evidence

The units of user behavior evidence values are mostly different such as the total number of incorrect password inputs by users, the historical occurrence rate of user IP addresses, and the user's sensitive service time. Therefore, before calculating the user behavior trust value, it is necessary to further process the collected user history behavior evidence to convert the data within the scope. According to the dimensionless method, the types of user behavior evidence can be divided into numerical type, percentage type, and Boolean type.

The focus of this article is the evaluation of the user behavior trust value; this article does not elaborate on the acquisition of evidence. This article uses existing methods [32] to standardize the evidence. After standardized processing, the behavior evidence set $Data = \{d_1, d_2, \dots, d_m\}$ from the user's access process can be obtained.

5.3. Weight of User Behavior Evidence

Determining the weight coefficient of each evaluation index is one of the keys to user behavior trust evaluation. The weight reflects the relative importance of each evaluation index. It should be noted that when the evaluation object and evaluation index are determined, the user's comprehensive trust value will be completely dependent on the value of the weight coefficient. Therefore, the rationality of the weight coefficient directly affects the rationality of the evaluation results and even affects the correctness and credibility of the conclusions. At present, there are many methods for calculating weight coefficients, mainly including subjective weighting methods and objective weighting methods.

5.3.1. Objective Weight

Among objective weighting methods, commonly used methods include the entropy weight method, the standard deviation method, and the CRITIC method [33]. Among them, the entropy weight method is a method that uses the concept of entropy to determine the index weight. In information theory, entropy is a measure of the uncertainty of the system state. The starting point of the entropy method involves reflecting the degree of importance of a certain index based on the degree of difference between the observed values of the same index. The applicability of this method is more extensive, and the requirements for the evaluation index are not high. The idea of the standard deviation method is very similar to the entropy weight method, but it is no longer based on information entropy, but on standard deviation. The CRITIC method comprehensively measures the objective weight of indicators based on the contrast strength of evaluation indicators and the conflict between indicators. The contrast intensity is expressed by the standard deviation. If the data standard deviation is larger, the fluctuation is greater, and the weight will be higher. The conflict is expressed by the correlation coefficient. If the correlation coefficient between the indicators is larger, the conflict is smaller, and then its weight is also lower. Therefore, the CRITIC method has higher requirements for the correlation between the evaluation index data.

Considering the randomness of network user behavior, abnormal user behavior can be regarded as a manifestation of disorder and abnormality in the process of user interaction with the system. Therefore, this paper chooses the entropy weight method to calculate the objective weight [34], and the specific process is as follows:

First, δ_{ij} represents the i -th behavior evidence of the user's j -th behavior, and then $\sum_{j=1}^n \delta_{ij}$ is the sum of the previous n evidences of the user. Therefore, the proportion of single behavioral evidences in the previous n times can be calculated:

$$P_{ij} = \delta_{ij} / \sum_{j=1}^n \delta_{ij} \text{ and } i = 1, 2, \dots, m, j = 1, 2, \dots, n \quad (2)$$

Next, calculate the entropy value of user behavior evidence e_{ij} :

$$e_{ij} = -K \sum_{j=1}^n P_{ij} \ln P_{ij} \text{ and } K = 1/\ln n \quad (3)$$

Finally, the entropy weight value of each user behavior evidence obtained through the entropy value is α_i , and the calculation formula is:

$$a_i = (1 - e_i) / (m - \sum_{i=1}^m e_i) \quad (4)$$

It can be seen from Equation (4) that α_i is a function that decreases as the entropy value increases. Through the above formula, we conclude that the entropy weight of user behavior evidence without abnormality is small, and the entropy weight of user behavior evidence with a greater degree of abnormality is greater. In addition, in order to facilitate subsequent calculations, we normalize the entropy weight of Equation (4) to obtain the objective weight set in the user behavior as $\omega_{OBi} = \{\omega_{OB1}, \omega_{OB2}, \dots, \omega_{OBm}\}$, in which ω_{OBi} represents the weight value of the i -th user behavior evidence in the objective weight set.

5.3.2. Subjective Weight

Because AHP can divide various factors within complex issues into interconnected and orderly levels to make them organized and based on the subjective judgment structure of a certain objective reality (mainly a pairwise comparison), the objective judgment results are directly and effectively combined to quantitatively describe the importance of the pairwise comparison of elements at a level. Therefore, this article used AHP to calculate the subjective weight of user behavior evidence [35]. The specific steps were as follows:

First, we established an AHP hierarchical model of user behavior. The model was divided into three levels: the target layer, the attribute layer, and the evidence layer. The target layer refers to user behavior; the attribute layer includes basic attributes, activity attributes, and security attributes; and the evidence layer refers to specific behavior evidence.

Next, for the user behavior evidence under different attribute levels, a 9-point system was used to compare the two factors to construct a judgment matrix, and the obtained judgment matrix was column-normalized. The specific rules of the nine-point system are shown in Table 1.

Table 1. Nine-point system pairwise comparison.

Scale	Definition and Description
1	Two elements have the same importance for an attribute.
3	Comparing two elements, one element is slightly more important than the other.
5	Comparing two elements, one element is obviously more important than the other.
7	Comparing two elements, one element is more important than the other.
9	Comparing two elements, one element is extremely more important than the other.
2, 4, 6, 8	The middle values of the above two judgments (1 and 3, 3 and 5, 5 and 7, 7 and 9)

The subjective weight set of all user behavior evidence in the user behavior trust evaluation is $\omega_{SUi} = \{\omega_{SUi1}, \omega_{SUi2}, \dots, \omega_{SUi m}\}$. Among them, $\omega_{SUi} = Z_{kl} * S_k$ represents the weight value of the i -th user behavior evidence in the subjective weight set (Z_{kl} is the weight value of each user behavior evidence, k is the attribute item, i is the user behavior evidence, and S_k is the weight value of each attribute item).

5.3.3. Integration Weight

In order to make the evaluation results more reasonable, based on the subjective ω_{SUi} and objective weights ω_{OBi} that were obtained, it was necessary to use optimization tech-

niques to construct an integrated weight model that reflected both subjective information and objective weights:

Let the integrated weight set be $\omega_{INi} = \{\omega_{IN1}, \omega_{IN2}, \dots, \omega_{IN3}\}$, where ω_{INi} represents the weight value of the i -th user behavior evidence in the integrated weight set. In order to analyze the influence and relationship between objective factors and subjective factors through the weight value in the set, we first determine the integrated weight. The sum of the square of the difference after subtracting the subjective weight and the objective weight for each item is the smallest; that is, $\sum_{i=1}^m (\omega_{INi} - \omega_{OBi})^2$ and $\sum_{i=1}^m (\omega_{INi} - \omega_{SUi})^2$ are the smallest, so the following optimization model can be constructed:

$$\min z = \sum_{i=1}^m [\alpha(\omega_{INi} - \omega_{OBi})^2 + \beta(\omega_{INi} - \omega_{SUi})^2] \quad (5)$$

Among them, α and β are a given constant, where α reflects the degree to which the system administrator's personal decision-making focuses on objective factors, and β reflects the degree to which the system administrator's personal decision-making focuses on subjective factors.

The overall user behavior trust evaluation value for user behavior is:

$$\min z = \sum_{i=1}^m [\alpha(\omega_{INi} - \omega_{OBi})^2 + \beta(\omega_{INi} - \omega_{SUi})^2] \quad (6)$$

For the security of the system, the trust evaluation of user behavior should be as strict as possible; that is, the overall user behavior trust evaluation value of user behavior should be the smallest. Therefore, construct the Lagrange function [36] according to Equations (5) and (6):

$$\begin{aligned} & F(\omega_{IN1}, \omega_{IN2}, \dots, \omega_{INm}, \lambda) \\ &= \sum_{i=1}^m [\alpha(\omega_{INi} - \omega_{OBi})^2 + \beta(\omega_{INi} - \omega_{SUi})^2] \\ & \quad + \sum_{j=1}^n \sum_{i=1}^m b_{ij} \omega_{INi} - 2\lambda \left(\sum_{i=1}^m \omega_{INi} - 1 \right) \end{aligned} \quad (7)$$

In the above formula, λ represents Lagrangian multipliers, and let $\frac{\partial F}{\partial \omega_{INi}} = 0$ and $\frac{\partial F}{\partial \lambda} = 0$; through simplification, we can get the following equations:

$$\begin{cases} (\alpha + \beta)\omega_{INi} - \lambda = \alpha\omega_{OBi} + \beta\omega_{SUi} - \frac{1}{2} \sum_{j=1}^n b_{ij} \\ \sum_{i=1}^m \omega_{INi} = 1 \end{cases} \quad (8)$$

If we calculate Equation (7), we can get:

$$\begin{aligned} \omega_{INi} &= \frac{\alpha}{\alpha + \beta} \omega_{OBi} + \frac{\beta}{\alpha + \beta} \omega_{SUi} \\ & \quad + \frac{1}{2(\alpha + \beta)} \left(\frac{1}{m} \sum_{j=1}^n \sum_{i=1}^m b_{ij} - \sum_{j=1}^n b_{ij} \right) \end{aligned} \quad (9)$$

$$b_i = \frac{1}{m} \sum_{j=1}^n \sum_{i=1}^m b_{ij} - \sum_{j=1}^n b_{ij} \quad (10)$$

We can further simplify Equation (9):

$$\omega_{INi} = \frac{1}{\alpha + \beta} \left(\alpha\omega_{OBi} + \beta\omega_{SUi} + \frac{1}{2}b_i \right) \quad (11)$$

Finally, we can get the integration weight ω_{INi} .

Theorem 1. Let $\omega_{OBi} \geq 0$, $\omega_{SUi} \geq 0$, and $\sum_{i=1}^m \omega_{OBi} = 1$, $\sum_{i=1}^m \omega_{SUi} = 1$; next, $(c = \alpha + \beta) \geq 1$, so that ω_{INi} in Equation (11) satisfies $\omega_{INi} \geq 0$ and $\sum_{i=1}^m \omega_{INi} = 1$.

When certifying, according to Equations (10) and (11), we can get:

$$\begin{aligned} \sum_{i=1}^m \omega_{INi} &= \frac{1}{\alpha+\beta} \left(\sum_{i=1}^m \omega_{OBi} + \sum_{i=1}^m \omega_{SUi} + \frac{1}{2} \sum_{i=1}^m b_i \right) \\ &= \frac{1}{\alpha+\beta} \left(\sum_{i=1}^m \omega_{OBi} + \sum_{i=1}^m \omega_{SUi} + \frac{1}{2} \left(m * \frac{1}{m} \sum_{j=1}^n \sum_{i=1}^m b_{ij} - \sum_{i=1}^m \sum_{j=1}^n b_{ij} \right) \right) \\ &= \frac{\alpha}{\alpha+\beta} + \frac{\beta}{\alpha+\beta} = 1 \end{aligned}$$

Then, we begin to prove that $\omega_{INi} \geq 0$ is established.

First, let $c = 1$, which is $\alpha, \beta \in [0, 1]$ and $\alpha + \beta = 1$. Put it into Equation (11) to calculate. If all $\omega_{INi} \geq 0$ holds, the conclusion is correct; otherwise, let $\omega_{INi1}, \omega_{INi2}, \dots, \omega_{INit} < 0$ because $\alpha, \beta \geq 0$ and $\omega_{OBi} \geq 0$, $\omega_{SUi} \geq 0$, and $b_i (i = 1, 2, \dots, m)$ are constants, so for any $INik (k = 1, 2, \dots, t)$, there is $c_k > 1$, making $\alpha\omega_{OBik} + \beta\omega_{SUi} + \frac{1}{2c_k}b_{ik} \geq 0$ hold, that is:

$$\frac{1}{c_k} \left(c_k * \alpha\omega_{OBik} + c_k * \beta\omega_{SUi} + \frac{1}{2}b_{ik} \right) \geq 0.$$

Take

$$c = \max\{c_1, c_2, \dots, c_t\} \quad (12)$$

and let

$$a' = c * \alpha, \beta' = c * \beta \quad (13)$$

Next, replace α, β with α', β' and put it into Equation (11):

$$\omega_{INi}' = \frac{1}{\alpha' + \beta'} \left(\alpha' \omega_{OBi} + \beta' \omega_{SUi} + \frac{1}{2} b_i \right) \geq 0 \quad (14)$$

Finally, c is the final result.

In summary, here are the steps to determine the weight:

Step 1. The system administrator gives the degree of bias toward subjective and objective weights $\alpha, \beta \in [0, 1], \alpha + \beta = 1$;

Step 2. Use Equation (11) to calculate ω_{INi} ;

Step 3. If all $\omega_{INi} \geq 0$, the calculation ends; otherwise, use Equations (12) and (13) to calculate c, α', β' ;

Step 4. Use Equation (14) to calculate ω_{INi}' .

5.3.4. Direct Trust Value Based on User Behavior

According to the user behavior evidence $Data = \{d_1, d_2, \dots, d_m\}$ obtained in Section 5.2 and the weight of evidence $\omega_{INi} = \{\omega_{IN1}, \omega_{IN2}, \dots, \omega_{INm}\}$ calculated in Section 5.3, the user's direct trust value T_{direct} can be calculated:

$$T_{direct} = d_1 * \omega_{IN1} + d_2 * \omega_{IN2} + \dots + d_m * \omega_{INm} \quad (15)$$

6. Comprehensive Trust Evaluation and Experimental Analysis

6.1. Comprehensive Trust Evaluation

In Sections 4 and 5, the user's indirect trust value $T_{indirect}$ and direct trust value T_{direct} are calculated respectively. Following that, the comprehensive trust value of the user is:

$$T_{com} = a * T_{indirect} + b * T_{direct} \quad (16)$$

Among them, $a, b (a, b \geq 0; a + b = 1)$ are given constants, where a reflects the degree to which the system administrator's trust evaluation of the user is more dependent on others' recommendations, and b reflects whether the system administrator's trust evaluation of the user is more dependent on the user's behavior. Under normal circumstances, the system administrator will use the user's access behavior as the main basis, or, that is, direct trust as the main basis, and indirect trust as the supplement. However, there are special circumstances. If a user's direct trust value deviates greatly from the historical trust value, the system administrator can learn from the indirect trust value to determine whether the reason for this situation was due to the user's violation or if there was a normal behavior different from historical behavior. Therefore, the values of a, b are dynamically set by the system administrator.

6.2. Experimental Analysis

Our experimental data came from a book e-commerce system that integrated purchases, downloads, and queries developed by our research group. Our system captured the user's mouse actions by embedding JavaScript code in the JSP page such as the user's input of a username. The number of backspaces, etc., and some basic configuration information of the computer used by the user can also be obtained through JavaScript such as screen resolution, browser type, etc. In this way, the system obtained the user's behavior sequence during the visit cycle. When the user launched the system, the system concentrated all user behavior sequences into the session listener class for unified processing and then used the trust evaluation model we built to get the user's behavior trust value.

We invited 30 volunteers to help us complete the experiment. These 30 volunteers visited our platform according to their needs within a month, so that our system backend stored a large number of real user behavior access records, and we chose from them. Ten types of behavior records were used as our behavioral evidence, and finally, we used our trust evaluation model to calculate the user's trust evaluation value from this behavioral evidence. The 10 types of behavioral evidence were as follows:

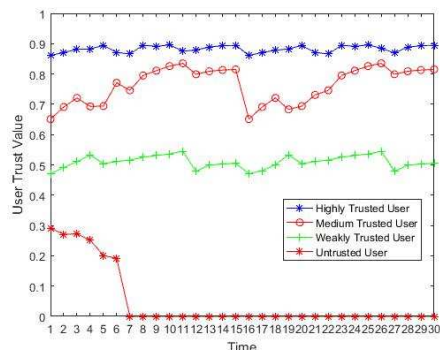
- (1) The historical appearance rate of the version of the user's login computer system (the rate of the version of the computer operating system such as Windows 7 or Windows XP, etc., used by the user when logging in to the experimental system, appearing in a specified number of consecutive user behaviors);
- (2) The historical appearance rate of the user's browser type such as IE or Chrome, etc. (the ratio of the browser used by the user to log in to the system in a specified number of consecutive user behaviors);
- (3) The IP historical appearance rate (the ratio of the IP address corresponding to the user logging in to the system in this behavior that appeared in the specified consecutive user behaviors);
- (4) The historical occurrence rate of geographic location (the rate of the specific geographic location such as school, home, library, etc., where the user logged into the system in a specified number of consecutive user behaviors);
- (5) The number of incorrect password inputs;
- (6) The number of incorrect usernames entered;
- (7) The total purchase value (whether the total purchase value of users exceeded the specified threshold);
- (8) The number of purchases of books (whether the number of purchases by the user exceeded the prescribed threshold);
- (9) The number of downloads of books (whether the number of downloads by users exceeded the prescribed threshold);
- (10) Access to sensitive services (the number of sensitive pages such as the password modification page, etc., that the user passed through during this visit was obtained by recording the URL path).

On the other hand, we set the trust level according to the security requirements of the system, as shown in Table 2 below.

Table 2. Classification of user behavior trust levels.

Trust Level	Ranges	Evaluation Result
H-level	(0.85~1)	Highly trusted user
M-level	(0.6~0.85)	Medium trusted user
Low	(0.3~0.6)	Weakly trusted user
E-low	[0~0.3)	Untrusted user

Through this system, we obtained the trust evaluation values of different types of users when accessing the network system within one month, as shown in Figure 3. The four curves shown in Figure 3 represent the changes in the trust values of users with different trust levels within one month. Because the model we proposed combined indirect trust value calculations, when the user accessed the system for the first time, we use the recommended trust value of the service provider as the initial trust value of the user. As shown in the figure, we could see that the initial trust value of each type was basically consistent with the trust level to which the user belonged. Secondly, we could also see that the user trust value calculated by our proposed model conformed to the principle of “slow increase and fast decrease”. As shown in the figure, if a middle-trust user has malicious behavior on the 15th day, the user’s trust value declines rapidly, and then there is a slow upward trend from the 16th day, even if the user’s behavior is well-documented thereafter. This can effectively prevent untrusted users from quickly increasing their trust value through deception. In addition, as shown in the figure, we found that the trust value of an untrusted user was 0 after the 7th day. This was because the user’s comprehensive trust value on the 6th day was already very low at 0.2, which was significantly lower than the threshold set by the system. Therefore, the system no longer allowed the user to access it.

**Figure 3.** Change curve of trust values of 4 different types of users.

Therefore, under the limitations of the proposed model, neither trusted users nor untrusted users can exhibit excessive malicious behavior; otherwise, the user will have to pay a certain price to make up for their fault.

In order to show the advantages of our proposed model, we compared our model with a type of model that only contained direct trust evaluation and continued to use the data of weakly trusted users and untrusted users to calculate the changes in the trust value of these two types of users, as shown in Figure 4. We found that in the model that only contained the trust evaluation method, because there was no indirect trust evaluation, the user’s initial trust value could only be set to a “relatively safe” value, which was 0.4. The disadvantage of this setting was that for users, because the initial trust value was lower than the user’s true trust value, this affected the permissions that the user could obtain; for untrusted users, because the initial trust value was higher than the user’s true trust value, this enabled them to successfully bypass the trust evaluation. The secondary verification increased the number of accesses to the system, which seriously endangered the resources

in the confidential system. Therefore, this model, which does not consider indirect trust assessment, is obviously unreasonable.

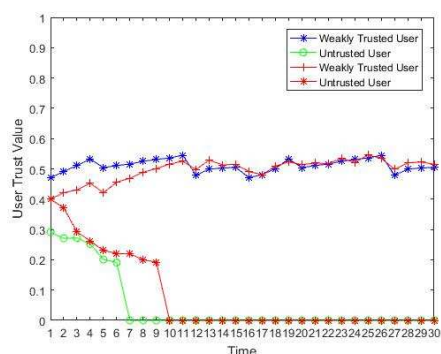


Figure 4. Comparison of the change curves of trust values between weakly trusted users and untrusted users.

In our method, the initial trust value of weakly trusted users was 0.5, and the initial trust value of untrusted users was 0.3. Following that, as the user interacted with the system, the user's trust value changed. We found that the trust value of untrusted users obtained by our proposed method decayed faster than in the other method. This was because in our method, we fully considered the impact of the recommended trust value on the user's final trust value, making the evaluation result closer to the real situation.

7. Conclusions and Recommendations

With the rapid development of the network environment, the problem of network information security has always been a hot issue for researchers. Because the behavior of network users is one of the important factors that affects network security, it is very important to conduct trust assessments of users. When evaluating the trust of network users, first, obtain complete user behavior evidence and balance the impacts of subjective and objective factors on the evaluation results; in addition, the impact of recommendation trust on the evaluation results must be emphasized as an important part of the evaluation. We designed a new trust evaluation model under the guidance of this idea. Through theoretical analysis and simulation experiments, our model was more efficient and reasonable than the existing common trust models.

In addition, our research also has certain limitations. First, the preprocessing of user behavior evidence is the basis of trust evaluation research. How to preprocess the acquired behavior evidence more reasonably is one of the contexts of future research; secondly, the division of user behavior evidence sets also plays a key role in the evaluation results; finally, we should also devote ourselves to applying our model to more fields, obtaining more behavioral evidence, and fusing these evidences from multiple sources to build a trust evaluation model based on the behavioral evidence of multi-source network entities.

Author Contributions: Y.Z., Z.W. (Zhigang Wang) assisted in collecting documents; Z.W. (Zenan Wu), L.T. designed the structure and conceived the idea of this paper; Z.W. (Zenan Wu) wrote the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Hebei Province Key R&D Program under Grant 19270318D, in part by the Hebei Internet of Things Monitoring Engineering Technology Research Center under Grant 3142018055, in part by the National Key R&D Program under Grant 2018YFC0808306, and in part by the Qinghai Province Internet of Things Key Laboratory Project under Grant 2017-ZJ-Y21.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Acknowledgments: The authors would like to thank all anonymous reviewers and editors for their helpful suggestions for the improvement of this paper.

Conflicts of Interest: The authors declare that there is no conflict of interest regarding the publication of this paper.

References

1. Ten, C.-W.; Manimaran, G.; Liu, C.-C. Cybersecurity for critical infrastructures: Attack and defense modeling. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2010**, *40*, 853–865. [\[CrossRef\]](#)
2. Dang-Pham, D.; Pittayachawan, S.; Bruno, V. Applications of social network analysis in behavioural information security research: Concepts and empirical analysis. *Comput. Secur.* **2017**, *68*, 1–15. [\[CrossRef\]](#)
3. Dang-Pham, D.; Pittayachawan, S.; Bruno, V. Investigation into the formation of information security influence: Network analysis of an emerging organisation. *Comput. Secur.* **2017**, *70*, 111–123. [\[CrossRef\]](#)
4. Wu, X.; Zhu, X.; Wu, G.-Q.; Ding, W. Data mining with big data. *IEEE Trans. Knowl. Data Eng.* **2013**, *26*, 97–107. [\[CrossRef\]](#)
5. Salah, T.A.; Albeshri, A.; Alsubhi, K. Integrating a high-reliability multicriteria trust evaluation model with task role-based access control for cloud services. *Symmetry* **2021**, *13*, 492.
6. Jiang, W.; Wang, G.; Bhuiyan, Z.A.; Wu, J. Understanding graph-based trust evaluation in online social networks. *ACM Comput. Surv.* **2016**, *49*, 1–35. [\[CrossRef\]](#)
7. Xiao, Y.; Pei, Q.; Liu, X.; Yu, S. A novel trust evaluation mechanism for collaborative filtering recommender systems. *IEEE Access* **2018**, *6*, 70298–70312. [\[CrossRef\]](#)
8. Wang, J.; Qiao, K.; Zhang, Z. Trust evaluation based on evidence theory in online social networks. *Int. J. Distrib. Sens. Netw.* **2018**, *14*, 1–10. [\[CrossRef\]](#)
9. Jiang, W.; Wu, J.; Li, F.; Wang, G.; Zheng, H. Trust evaluation in online social networks using generalized network flow. *IEEE Trans. Comput.* **2015**, *65*, 952–963. [\[CrossRef\]](#)
10. Yang, M.; Zhang, S.; Zhang, H.; Xia, J. A new user behavior evaluation method in online social network. *J. Inf. Secur. Appl.* **2019**, *47*, 217–222. [\[CrossRef\]](#)
11. Gong, Y.; Chen, L.; Ma, T. A comprehensive trust model based on social relationship and transaction attributes. *Secur. Commun. Netw.* **2020**, *2020*, 1–10. [\[CrossRef\]](#)
12. Ceolin, D.; Groth, P.; Nottamkandath, A.; Fokkink, W.; van Hage, W.R. *Analyzing User Demographics and User Behavior for Trust Assessment*; Vrije Universiteit Amsterdam: Amsterdam, The Netherlands, 2014; pp. 219–241.
13. Ghosh, I.; Chaudhuri, T.D. FEB-stacking and FEB-DNN models for stock trend prediction: A performance analysis for pre and post Covid-19 periods. *Decis. Mak. Appl. Manag. Eng.* **2021**, *4*, 51–86. [\[CrossRef\]](#)
14. Liu, Y.; Wang, Y.; Sun, X.Y. Research on behavior trust evaluation method of cloud services based on membership theory. *Appl. Mech. Mater.* **2013**, *427*, 2377–2382. [\[CrossRef\]](#)
15. Wang, J.; Jing, X.; Yan, Z.; Fu, Y.; Pedrycz, W.; Yang, L.T. A survey on trust evaluation based on machine learning. *ACM Comput. Surv.* **2020**, *53*, 1–36. [\[CrossRef\]](#)
16. Zhou, G.; Wang, K.; Zhao, C.; Zhou, G. A dynamic trust evaluation mechanism based on affective intensity computing. *Secur. Commun. Netw.* **2016**, *9*, 3752–3761. [\[CrossRef\]](#)
17. Li, L.; Feng, J.; Ye, H.; Liu, X. Trust research on behavior evaluation based on fuzzy similarity. *IEEE Access* **2020**, *8*, 204203–204213. [\[CrossRef\]](#)
18. Messinis, S.; Vosniakos, G. An agent-based flexible manufacturing system controller with Petri-net enabled algebraic deadlock avoidance. *Rep. Mech. Eng.* **2020**, *1*, 77–92. [\[CrossRef\]](#)
19. Lin, C.; Tian, L.; Wang, Y. Research on user behavior trust in trustworthy network. *J. Comput. Res. Dev.* **2008**, *45*, 2033–2043.
20. Zhang, S.-B.; Xu, C.-X. Study on the trust evaluation approach based on cloud model. *Chin. J. Comput.* **2013**, *36*, 422–431. [\[CrossRef\]](#)
21. Tian, L.; Lin, C. Evaluation mechanism for user behavior trust based on DSW. *J. Tsinghua Univ.* **2010**, *50*, 763–767.
22. Meng, X.; Ma, J.; Lu, D.; Wang, Y. Comprehensive trust evaluation model in social networks. *J. Commun.* **2014**, *35*, 136–143. [\[CrossRef\]](#)
23. Szpyrka, M.; Jasiul, B. Evaluation of cyber security and modelling of risk propagation with Petri nets. *Symmetry* **2017**, *9*, 32. [\[CrossRef\]](#)
24. Kai-Qing, Z.; Azlan, M.Z.; Li-Ping, M. Dynamic properties of fuzzy Petri net model and related analysis. *J. Cent. South Univ.* **2015**, *22*, 4717–4723.
25. Wai, R.-J.; Lin, Y.-W. Adaptive moving-target tracking control of a vision-based mobile robot via a dynamic Petri recurrent fuzzy neural network. *IEEE Trans. Fuzzy Syst.* **2012**, *21*, 688–701. [\[CrossRef\]](#)
26. Zhou, J.; Reniers, G. Modeling and application of risk assessment considering veto factors using fuzzy Petri nets. *J. Loss Prevent. Proc. Ind.* **2020**, *67*, 104216. [\[CrossRef\]](#)

27. Chen, Z.; Tian, L.; Lin, C. Trust evaluation model of cloud user based on behavior data. *Int. J. Distrib. Sens. Netw.* **2018**, *14*, 1–10. [[CrossRef](#)]
28. Buczak, A.L.; Guven, E. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Commun. Surv. Tutor.* **2015**, *18*, 1153–1176. [[CrossRef](#)]
29. Zhou, F.; Huang, W.; Zhao, Y.; Shi, Y.; Liang, X.; Fan, X. ENTVis: A visual analytic tool for entropy-based network traffic anomaly detection. *IEEE Eng. Med. Biol. Mag.* **2015**, *35*, 42–50. [[CrossRef](#)] [[PubMed](#)]
30. Sengottuvelan, P.; Lokeshkumar, R.; Gopalakrishnan, T. An improved session identification approach in web log mining for web personalization. *J. Internet Technol.* **2017**, *18*, 723–730. [[CrossRef](#)]
31. Shen, C.; Cai, Z.; Liu, X.; Guan, X.; Maxion, R.A. MouseIdentity: Modeling mouse-interaction behavior for a user verification system. *IEEE Trans. Hum. Mach. Syst.* **2016**, *46*, 734–748. [[CrossRef](#)]
32. Luor, D.-C. A comparative assessment of data standardization on support vector machine for classification problems. *Intell. Data Anal.* **2015**, *19*, 529–546. [[CrossRef](#)]
33. Mukhametzyanov, I. Specific character of objective methods for determining weights of criteria in MCDM problems: Entropy, CRITIC and SD. *Decis. Mak. Appl. Manag. Eng.* **2021**, *4*, 76–105. [[CrossRef](#)]
34. Al-Aomar, R. A combined AHP-entropy method for deriving subjective and objective criteria weights. *Int. J. Ind. Eng. Theory* **2010**, *17*, 12–24.
35. Wen, S.; He, Y.; Li, W.; Yang, R. Evaluation of trademark right based on AHP method and comprehensive fuzzy decision method. In Proceedings of the 2020 International Conference on Urban Engineering and Management Science (ICUEMS), Zhuhai, China, 24–26 April 2020; pp. 460–466.
36. Kim, D.S.; Son, T.Q. Some new properties of the Lagrange function and its applications. *Fixed Point Theory Appl.* **2012**, *2012*, 192. [[CrossRef](#)]