© 2016 IJSRSET | Volume 2 | Issue 3 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Fake Product Detection and Reputation System for E-commerce

Indu T R*, Ayana Joseph, Divya V R, Prof. Linda Sara Mathew

Department of Computer Science, M. G. University, Kerala, India

ABSTRACT

E-commerce is a new way to transact business electronically through networks and the Internet. During the last 5 years, the number of internet users rapidly increased which in turn led to fast growth in e-commerce market. The major drawback of e-commerce application is the lack of information about sellers behavior. Buyers are afraid of being cheated by the seller and often feel hesitate to make transaction at e-commerce site. While people are enjoying the benefits from online trading, criminals are also taking advantages to conduct fraudulent activities against shonest parties to obtain illegal profit. Fraud detection is essential to prevent such illegal and fraud activities. In this paper, we propose an implementation algorithm for a new reputation system based on seller-buyer closeness degree for e-commerce. The proposed reputation system can be understood and implemented easily in real e-commerce industry.

Keywords: Reputation, Seller, Buyer, Closeness, Algorithm, Fraud Detection

I. INTRODUCTION

Presently, many people use internet so that the e-commerce transaction grow so fast. E-commerce not only covers the activity of buying and selling goods or services, but also covers the activities of the lease or auction goods. Nonetheless, e-commerce application has a crucial problem while it is developing very rapidly. That problem is trustworthiness problem. That problem appears because buyers usually feel very doubtful as to whether seller behavior will appropriate his expectations. Generally buyers are not living together or never met before with the sellers so that those conditions have a chance to be online fraud. In the absence of an indicator or parameter of the sellers behavior on e-commerce applications, then during that time the buyer will make gambling to choose the right seller. Consequently, buyers often feel hesitate to make transaction at e-commerce site because they are afraid of being cheated by the seller.

Starting from problems above, present researchers who study on e-commerce are developing a reputation system for sellers. Reputation is someone's estimate of an individual with respect to his personality or other qualities, after some direct or indirect interaction with the individual. So, almost all research in reputation system relies on feedback rating that is obtained from system users.

II. METHODS AND MATERIAL

1. Related Work

Reputation system helps buyers to choose right sellers who have minimal chance to cheat. Feedback or satisfaction score is given by the buyer after each transaction with the seller will help to assess the reputation of a seller. The average of the value of the buyers satisfaction will give the reputation score of the seller. Thus, the reputation system depends on feedback rating which is obtained from system users.

A recent development in this field is the seller-buyer closeness degree-based reputation system. A new metric called seller-buyer closeness degree was proposed by this system helps to strengthen the feedback-based reputation system. Social media is used as a tool in this metric to assess the weight of seller-buyer relationship. Besides the above mentioned metric the system used five more metric to build a complete reputation system called seller-buyer closeness degree based reputation system. The five metrics used are feedback score metric, number of transaction, transaction price, time decay coefficient, closeness degree between buyer and seller.

The feedback score metric is the main part of reputation system. It relies on buyer's evaluation after transaction with seller. Feedback score metric can be divided into five classes of evaluation value. They are very dissatisfied (-4), dissatisfied (-2), normal (0), satisfied (1), very satisfied (2). The point of each evaluation value is indicated in the parenthesis. For a buyer who transacts with seller s in the ith transaction is given by $Rt_i(s)$.

$$Rt_i(s)=\begin{cases} 2 \\ 1 \\ 0 \\ -2 \\ -4 \end{cases} \qquad (1)$$

The number of transaction metric is used as the denominator of the first metric so that we get the average of feedback score. The number of transaction is indicated by N_s. The third metric is transaction price. Buyers who transact in large price usually make mature assessment and their feedback will be more precious so that they can avoid the loss. The transaction price metric can be symbolized as follow.

$$TP_i=\frac{P_i}{\bar{P}} \qquad (2)$$

Here the symbol TP_i is the metric transaction price, P_i is the price of ith transaction and P is the average of all transaction prices. TP_i can be considered as the ratio of the transaction price in the ith transaction against the average of all transaction prices.

Time decay coefficient is the fourth metric that is used to build a dynamic and non-monotonic reputation model. In the following formula, d_{ti} is time decay coefficient of ith transaction, w_c is current time week, and w_i is time week when ith transaction occurred.

$$d_{ti}=2^{-0.1(w_c-w_i)} \qquad (3)$$

Finally, we have seller-buyer closeness degree metric, which is important because in order to distinguish the knowledge of buyer about the seller, we give different weight feedback for different buyer. Social network is used as a tool to find the closeness degree of e-commerce users. The above metric can be symbolized as

$$CD_i(s,b)=\begin{cases} 1+\frac{|F_s\cap F_b|}{|F_s\cup F_b|} & , \ |F_s\cup F_b|\geq 100 \\ 1 & , \ |F_s\cup F_b|<100 \end{cases} \qquad (4)$$

Here $CD_i(s,b)$ is the closeness degree between seller s and buyer b during ith transaction. F_s is the friend set of seller s, F_b is the friend set of buyer b, whereas $|F_s\cap F_b|$ is sum of friend which become friend of s and b, and $|F_s\cup F_b|$ is sum of friend set which become friend of s or b.

All the above five metrics are used to produce a complete reputation system which is obtained by a multiplication formula between these metrics as shown below.

$$R(s)=\frac{1}{N_s}\sum_{i=1}^{N_s}(R_{ti}(s)\times TP_i \times d_{ti} \times CD_i(s,b)) \qquad (5)$$

where:

R(s) - the average of reputation score of the seller s,
N_s - the number of transaction which seller s make,
$R_{ti}(s)$ - rating (feedback score) which given by the buyer who transacts with seller s in the ith transaction,
TP_i - the ratio of the transaction price in the ith transaction against the average of all transaction prices,
d_{ti} - time decay coefficient of ith transaction, and
$CD_i(s,b)$ - closeness degree between seller s buyer b when the ith transaction occurred.

2. Proposed System

We develop algorithm from the reputation system concept that has discussed in the previous section. The reputation algorithm can be divided into four functions or procedures. Those are *initial(), getClosenessDegree(), updateDaily(), newTransaction().* In order to save important data in reputation system we proposed several global variables.

The variables can be seen as follow

TABLE I
GLOBAL VARIABLES OF REPUTATION ALGORITHM

Variables	Function
TR=[][]	Set of reputation score from all sellers in every transaction. Index of first dimension of *TR* is called *s* as *seller id*. Then, index of second dimension of *TR* is called *i* as transaction *id* of seller *s* and the second dimension contain transaction reputation score of seller *s* when he has done *i*-th transaction. We assume the *seller id* and *seller id's transaction* start sequentially from 1. When declared, all sellers id and transactions id in e- commerce database has saved into index of first and second dimension of this variable.
R=[]	Array of final reputation score from all sellers based on his every transaction reputation score in array *TR*. Index of array *R* is called *s* as *seller id* and this array contain final reputation score of seller *s*. When declared, all sellers id in e-commerce database has saved into index of this variable.
dbTransact	E-commerce database input, consisting of *idSeller*, *idTtransaction*, *rating*, *idBuyer*, *price*, and *time* columns.

A. Initial Procedure

Above global variables are used in the four functions or procedures of reputation algorithm. *initial* procedure is the first one. When our reputation system is first installing in e-commerce application this procedure is executed. All transaction data, such as *idSeller*, *idTransaction*, *rating*, *idBuyer*, *price*, and *time*.

Algorithm 1 Initial Reputation Algorithm

```
procedure initial
    avePrice ← calAvePrice(dbTransact)
    for s =1to TR.length do
        tempSum ← 0
        for i =1to TR_s.lengthdo
            Rt← getRating(dbTransact,s,i)
            TP ← getPrice(dbTransact,s,i)
            TP ← TP/avePrice
            t ← getTime(dbTransact,s,i)
            diff ← getDiffTime(getCurTime(),t)
            d ← 2−0.1∗diff
            idBuyer ← getIdBuyer(dbTransact,s,i)
            CD← getClosenessDegree(s,idBuyer)
            TR_si ← Rt∗TP∗CD
            tempSum ← tempSum + TRsi ∗d
        end for
        R_s← tempSum/TRs.length
    end for
end procedure
```

In the above procedure our aim is to calculate reputation score of all current sellers in e-commerce's database. Before calculating it this procedure will compute the average price from all transaction which is needed by transaction price metric. In order to calculate the seller's reputation score in every transaction we iterate all sellers in database and his transactions. Since e-commerce application has saved all needed information for each transaction, such as rating from buyer to seller, price, transaction time, and seller and buyer user id in facebook, this procedure get important value from e-commerce database. The average reputation score is calculated by dividing the totalled seller's reputation score by the number of transactions.

B. Closeness Degree Function

The *getClosenessDegree* is the second function. In order to calculate the index of closeness degree between certain buyer and seller from *facebook,* this function is executed by other function or procedure in reputation algorithm. *idSeller* and *idBuyer* are the two variables used in this function.

Algorithm 2 Complete Closeness Degree

```
function getClosenessDegree(idSeller, idBuyer)
    import:fbAPI   -fbAPI is facebook library
    const:friendTreshold ← 100
    mutual ← fbAPI.getMutual(idSeller,idBuyer)
    fS← fbAPI.getTotalFriend(idSeller)
    fB← fbAPI.getTotalFriend(idBuyer)
    union ← fS+ fB−mutual
    ifunion < friendTresholdthen
        return 1
    else
        return 1+(mutual/union)
    end if
end function
```

The above function is the implementation of the closeness degree between buyer and seller metric. All buyer's and seller's *user id* in *facebook* are stored in e-commerce application. It will help in the retrieval of required data. This function need facebook API in order to connect to facebook web application and get some user information, such as total of mutual friends between seller and buyer, and total of seller friend and buyer friend.

C. Update Daily Procedure

Everyday this procedure will be executed by the server of e-commerce application. It is the implementation of the fourth metric, time decay. Since we update the reputation score of each seller in everyday the above concept is used to build a dynamic and non-monotonic reputation model. Reputation score of each seller varies depending on the selling performance.

Algorithm 3 Update Daily Reputation Algorithm

procedure *UpdateDaily*
for *s* =1**to***TR.length* **do**
  $tempSum \leftarrow 0$
  for *i* =1**to***TRs.length* **do**
    $t \leftarrow getTime(dbTransact,s,i)$
    $diff \leftarrow getDiffTime(getCurTime(),t)$
    $d \leftarrow 2^{-0.1*diff}$
    $tempSum \leftarrow tempSum + TR_{si}*d$
  end for
  $R_s \leftarrow tempSum/TR_s.length$
end for
end procedure

The above procedure will iterate all seller and each of its transaction history. In each transaction the difference between current time week and transaction's time week of each seller will be checked. This difference use week unit and it is used to calculate time decay coefficient. After multiplying this coefficient with the reputation value of that transaction it is saved in the variable *tempSum*. This variable will be totalled and divided by total transaction made by the seller s. The final reputation score of seller s for each day will be calculated and saved in array R.

D. *NewTransaction Procedure*

The newTransaction procedure is executed when new transaction is done by a seller is done. The new transaction information is recorded first by the e-commerce database. In order to fetch transaction information in e-commerce database, after executing this procedure the e-commerce application request input variable *s* as seller id who make a new transaction, and variable *i* as transaction id.

Algorithm 4 New Transaction

procedure *NewTransaction(s,i)*
  $avePrice \leftarrow calAvePrice(dbTransact)$
  $Rt \leftarrow getRating(dbTransact,s,i)$
  $TP \leftarrow getPrice(dbTransact,s,i)$
  $TP \leftarrow TP/avePrice$
  $t \leftarrow getTime(dbTransact,s,i)$
  $diff \leftarrow getDiffTime(getCurTime(),t)$
  $d \leftarrow 2^{-0.1*diff}$
  $idBuyer \leftarrow getIdBuyer(dbTransact,s,i)$
  $CD \leftarrow getClosenessDegree(s,idBuyer)$
  $TR_{si} \leftarrow Rt*TP*CD$
  $R_s \leftarrow (R_s*TR_s.length + TR_{si}*d)/TR_s.length$
end procedure

The *newTransaction* procedure is almost similar to initial procedure. This procedure calculate reputation score of seller *s*' new transaction based on the reputation system where this procedure compute feedback score metric, transaction price metric, time decay metric and closeness degree metric. In order to produce reputation score of new transaction all metrics values except time decay metric value are multiplied and saved in table R_{si}. Since time decay metric has the nature of dynamic score builder it is not included and will be used only when the final seller reputation score is calculated.

III. RESULTS AND DISCUSSION

IMPLEMENTATION PROCEDURE

For better understanding about the seller-buyer closeness degree-based reputation algorithm, we show the way to use it. It is to be noted that for an e-commerce developer, this reputation system should be implemented as a library or plugin. For maintaining the code of e-commerce application and reputation system easily, it aims to separate source code. In most cases the programmer team of e-commerce application will be different from the reputation system because of perspective differences. The source code of e-commerce application need not be reorganized by the programmer team if there is an improvement. Also, whenever the developers want to change this reputation system with another, they just need to install other library or plugin without changing the main code of e-commerce application.

Initially for implementing the reputation algorithm into a plugin, the developer must make sure that the e-commerce application is customer-to-customer type. All sellers will be checked and a score will be provided. In some cases where the e-commerce application is concerned only with one seller this reputation system is not useful because the buyer cannot compare one seller with others. Then, the developer must ensure that the e-commerce application provide all information that is needed by our reputation system, such as rating point after transaction successful (for feedback score metric), price of transacted product (for transaction price metric), time when the transaction occurred (for time decay metric), and seller's and buyer's id or username in facebook in order to get their closeness degree value.

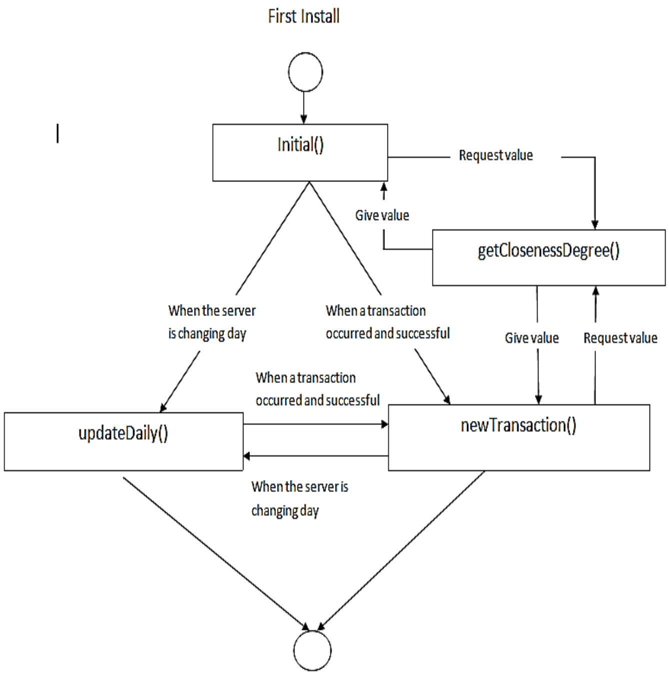

Figure 1. Algorithm Diagram

For each missing information, the developers must provide a related functionality to obtain such information from the user. Database should be provided in order to save global variables S and R. If the code is free from all bugs and if it is ready to run, follow the instructions shown below.

- The developer must run *initial* procedure from ecommerce application only for the first time in order to generate all sellers' reputation score.
- After that, write a brief code in the e-commerce application to execute *updateDaily* procedure when the server of e-commerce application by day.
- Then, write a brief code in the e-commerce application to execute *newTransaction* procedure when new transaction is done by a seller.
- Finally, write a brief code in commerce application to show value of first dimension in global reputation variable (*Rs*) when the buyer open sellers list in e-commerce application. This *Rs* contain final reputation score of all sellers in certain day.

IV. CONCLUSION

In seller-buyer closeness degree-based reputation system, an algorithm to compute reputation score is proposed. The proposed algorithms consisted of 4 functions and procedures and are used together as reputation system. By using the reputation system, buyers can select and choose the honest seller to transact so that the deception action in e-commerce application can be minimized. This model can be used to detect fraud products and sellers and considerably reduce the fraudulent activities in e-commerce.

V. REFERENCES

[1] I Putu Arya Dharmaadi Suhardi, "Implementation of Seller-Buyer Closeness Degree-based Reputation System for E-commerce",International Conference On Information Technology Systems And Innovation (ICITSI) ,November 2014
[2] F. Wu, H.-H. Li, and Y.-H. Kuo, "Reputation evaluation for choosing a trustworthy counterparty in C2C e-commerce," Electronic Commerce Research and Applications, vol. 10, no. 4, pp. 428–436, Jul. 2011.
[3] Q. Liang and X. Gan, "Research on Trust Evaluation Algorithm for E-Commerce Based on Reputation," 2011 International Conference on Business Computing and Global Informatization, pp. 39–42, Jul. 2011.
[4] I. P. A. Dharmaadi, Suhardi, F. Hidayat,"Reputation system based on seller-buyer closeness degree for e-commerce," International Conference on ICT for Smart Society, 2014.