A New Feedback-Analysis based Reputation Algorithm for E-Commerce Communities

Hasnae Rahimi
Information Security Research Team (ISeRT), University Mohammed V, ENSIAS, Avenue Mohamed Ben Abdellah Regragui, Rabat, Morocco
hasnae.rahimi@gmail.com

Hanan El Bakkali
Information Security Research Team (ISeRT), University Mohammed V, ENSIAS, Avenue Mohamed Ben Abdellah Regragui, Rabat, Morocco
h.elbakkali@um5s.net.ma

Résumé

Traitant le contenu croissant fourni par les utilisateurs dans les applications de e-commerce, les Systèmes de réputation (TRS) sont largement utilisés en ligne pour calculer le degré de confiance de chaque produit tout en se basant sur les évaluations des clients. Cependant, les feedback des clients doivent être également utilisés par les TRS.
En conséquence, nous proposons dans ce travail une nouvelle architecture pour TRS en e-commerce, qui s'appuie sur l'analyse du texte des feedback pour calculer le score de réputation. Cette architecture est basée sur une couche intelligente qui propose à chaque utilisateur (c-à-d. «le fournisseur du feedback») ayant donné sa recommandation, une collection de feedback préfabriqués que l'utilisateur devrait aimer ou pas *(like/dislike)*. L'algorithme de réputation proposé calcule ainsi le degré de réputation de l'utilisateur, celui du produit et du feedback. Dans ce travail, nous présentons aussi un état de l'art des outils et algorithmes de *textmining* à utiliser pour produire les feedback préfabriqués et les classifier dans des catégories différentes.

Abstract

Dealing with the ever-growing content generated by users in the e-commerce applications, Trust Reputation Systems (TRS) are widely used online to provide the trust reputation of each product using the customers' ratings. However, there is also a good number of online customer reviews and feedback that must be used by the TRS.
As a result, we propose in this work a new architecture for TRS in e-commerce application which includes feedback' mining in order to calculate reputation scores. This architecture is based on an intelligent layer that proposes to each user (i.e. «feedback provider») who has already given his recommendation, a collection of prefabricated feedback to like or dislike. Then the proposed reputation algorithm calculates the trust degree of the user, the feedback's trustworthiness and generates the global reputation score of the product according to his «likes» and «dislikes». In this work, we present also a state of the art of text mining tools and algorithms that can be used to generate the prefabricated feedback and to classify them into different categories.

Mots-clés

La confiance, les systèmes de réputation, le e-commerce, les feedback textuels, le textmining, analyse de sentiment

Keywords

Trust, Trust Reputation Systems, e-commerce, textual feedback, text mining, sentiment analysis

1. Introduction

With the rapid growth of information available in e-commerce applications, users need to feel safe while purchasing a product or a service. In fact, in order to achieve an electronic transaction, they have to trust the product and its quality. That's the reason why users rate available products in the e-commerce application and leave their comments, opinions and textual feedback in the discussion forums, blogs etc.
Dealing with this rapid growth of information overload in e-commerce applications, Trust Reputation Systems (TRS) are widely used online to help customers find trustful products. In fact, TRS are Recommender systems based on intelligent algorithms that generate trust reputation score for a product.
Most popular TRS apply algorithms that collect opinions from customers in the form of ratings on products, services or service providers. In addition to the customer rating, there is also customer textual feedback or subjective reviews which express the customer's opinion and intention. This information must be used to generate a more precise public reputation of the product (Gutowska and Sloane, 2009).
In the literature, there are many works such as [2], (Liu, Zhang *et al.*, 2011), (Steinbrecher, Groß *et al.*, 2009), (Josang, Roslan *et al.*, 2007) that propose reputation algorithms for calculating a reputation or defining a specific set of possible reputations or ratings. However, few of them such as (Mármol, Girao *et al.*, 2010), (Jøsang, Quattrociocchi *et al.*, 2011) have been devoted to the semantic analysis of textual feedback in order to generate a trust score for the product and especially the trust degree of the user.
In contrast to these papers, our main contribution in this approach is to analyse the user's attitude toward specific prefabricated textual feedback. Our proposed TRS architecture aims to provide the user with the possibility to like or dislike – via a specific interface- some feedback summarizing several users' feedback in addition to fake and prefabricated feedback. This selection takes place after a user gives his appreciation (a numeric value) of the product within his textual feedback. Then the user is asked to validate his appreciation and feedback.
Our approach relies on an algorithm that includes semantic feedback analysis in order to generate the most reliable reputation score for a product since the feedback affects users' decisions more than numeric scores alone (Zhao and Yang *et al.* 2010), (Snijders and Matzat 2010). This proposed algorithm also calculates and updates the trust degree of the user after any participation in the TRS.
Besides, opinion mining algorithms are essential in the proposed architecture in order to extract from the customer's semantic feedback individual words, which are considered a unit of opinion information so as to provide the review sentiment. In fact, the text mining algorithm is needed to assign a numerical score to each customer's review to reflect its sentiment orientation (positivity, negativity and objectivity).
To sum up, we propose an algorithm to calculate the trust degree of the user based on the trustworthiness of the feedback he liked or disliked. The system proposes to the user a selection of prefabricated feedback and the user has to either like or dislike them like in the social network «Facebook». Some critics would say that even if the user is very trustworthy, his degree of trust may be enhanced by only a low increment because the feedback he has evaluated has low trustworthiness, and vice versa. In fact, this is the purpose: we can't determine if the user is very trustworthy. His trustworthiness evidence depends on his reaction and choices. Therefore, the trustworthiness of a feedback reflects the level of user's trustworthiness since his like/ dislike is valid for a trustful or distrustful feedback. The like/dislike strategy seems to be a technique that gives a binary result either trustful or distrustful feedback or user. However, it is not because the reputation algorithm generates many levels of user's trust degree. The like/dislike technique is a multi-level result technique that helps calculate the trust degree of the user in a very detailed dimension (see fig.3).
The remainder of this paper is structured as follows. In section 2 we review the terminology of trust and reputation systems. Section 3 presents some related TRS, in addition to a brief state of the art of text mining algorithms that can serve our purposes. The architecture and the algorithm related to our TRS are detailed in section 4. finally, we come up with some concluding remarks and an outlook on future work.

2. Trust and Reputation Background

Reputation and trust must be assigned closely to web content in order to estimate the usefulness of the web content and to use its trustworthiness. In fact, we need this estimation in order to help users react and interact with the content, share, comment and rate the content. We need to create social interaction and to develop further relations in order to collect additional information, then filter it and decide about its reliability using TRS. Consequently, users who are ready to evaluate the content can then rate it. They then become useful «reviewers». This rating, of course, is accompanied by a rating algorithm. In fact, the rating aims at generating a reputation. Then the assignment between the rating and the reputation must be done. A reputation algorithm calculates the reputation of the content from the ratings (Steinbrecher, Groß *et al.*, 2009)

2.1. Trust Definitions

In this work, we add textual feedback analysis in the calculus of the product reputation score. Initially, we will focus on some definitions used in the paper. We will start by highlighting some definitions of trust and reputation.

2.1.1. Definition 1

The definition of Trust is closely related to the willingness to pay the actual selling price in online markets (such as eBay), without considering the «underlying» trust of other buyers, sellers and even the product.
In fact, trust is also defined as the ability to rely on someone or something, to rely on its truthfulness and its strength to prove its reliability. In e-commerce, being trusted is a quality characterizing a product that a user claims to know either intuitively or from a past experience, which is more trustful, or because other users estimate that it is a reliable product (Steinbrecher, Groß *et al.*, 2009), (Josang, Roslan *et al.*, 2007).
However, trust which is not based on logical evidence corroborated by real experience and analytical examination is useless and does not help generate a reliable reputation.

2.1.2. Definition 2

Trust is also considered as a subjective evaluation of the potential outcomes and risks involved in relying on a partner. Indeed, trustworthiness refers to how much we can trust the product or the user's intervention concerning the product, so as to help any user develop his or her opinion and reputation image concerning the product.
However, user's intentions differ when sharing their «experience» by rating or writing a textual feedback (Steinbrecher, Groß *et al.*, 2009), (Zhao, Yang *et al.*, 2010). As a result, we should not trust users' honesty if it is shown through their interventions.

2.1.3. Definition 3

In general, trust can be described as the willingness of an agent to be vulnerable to another agent's interventions and be convinced by the actions of this agent based on previous expectations. Indeed, the other agent performance will be of particular importance to any potential user willing to deal with a future transaction. Irrespective ability to monitor or control the other party is a vulnerability that has to be changed to a convincing strength of a party that is able to give a structured and logical statement accompanied by well-built arguments and proof (Rahimi and El Bakkali, 2012), (Jøsang and Golbeck, 2009).
Actually, this trust is represented by rating and semantic feedback. Moreover, unreliable scores and feedback must be detected and examined with specific care in order to generate the most trustworthy reputation.
Tightly connected with the notion of trust is that of reputation, which could be seen as a collective, shared assessment of the same aspects.

2.2. Trust Reputation Systems Definitions

Reputation is generally said or believed to be about a person's or things' character or standing. Related to products and services, it is the subjective opinion based on feelings, past experiences and the viewpoint of a circle of "trustful" people. Reputation is often used in the sense of the community's general reliability and trustworthiness evaluation of a service entity (Josang, Roslan *et al.*, 2007), (Mayer, Davis *et al.*, 1995).
Therefore, this trust reputation needs to be gathered, collected and filtered in order to generate the most trustful reputation associated with a service, a product or a user.
To do that, Trust Reputations Systems are available tools that can work on trust reputation using algorithms with the purpose of calculating the most reliable evaluation.
We give hereafter some definitions of trust reputation systems.

2.2.1. Definition1: TRS towards the buyer, the seller and the whole community

Trust Reputation Systems (TRS) are an important class of decision support tools that can help reduce risk when dealing with transactions and interactions online. From the individual buyer's viewpoint, a TRS can help reduce the risk related to any particular interaction in the e-commerce application. From the service provider's viewpoint, it represents a marketing tool attracting more buyers and convincing them to purchase. From the community viewpoint, it represents an application of social interaction, moderation and control, as well as a method to assess trust by improving the quality of online markets and communities (Gutowska and Sloane, 2009), (Komiak, 2010).

2.2.2. Definition 2: Robustness of TRS as a decision-making tool in e-commerce

TRS represent a decision-making mechanism that allows parties to rate each other in order to give consumers a relatively clear idea of whether to go through with a transaction or not. They allow them to evaluate the reputation of a product, transaction, online merchant through the experience of other users. In virtual environments that apply reputation systems, users can decide whether to trust an online merchant based on the probable trust they have on the provider of the feedback (Ghazizadeh, Ehsaei *et al.*, 2012) Actually in e-commerce, users have to trust information from unknown and unreliable sources and anonymous users (Rahimi and El Bakkali, 2012). That's why, robust trust reputation systems are supposed to reduce the probability that a user is untrustworthy. Consequently, TRS reduce the probability that the user's interaction and input in the application is untrustworthy. Reducing that risk has an efficient impact only at the moment of the user's intervention. Then the robustness of a TRS depends on how much it reveals the truthfulness of the users' recommendations to one other in a specific time and context.

3. Related work

Many authors such as (Liu, Zhang *et al.*, 2011), (Steinbrecher, Groß *et al.*, 2009), (Josang, Roslan *et al.*, 2007), (Golbeck and Hendler, 2004), (Ghazizadeh and Hussin, 2012) propose in their work several TRS architectures with different algorithms to calculate the reputation score related to the product. Nevertheless, few academic works on TRS have been devoted to the inclusion of the semantic analysis of feedback in the calculus of the trust score of the product and especially the trust degree of the user. Even in studies attempting to provide more complex reputation methods such as (Josang, Roslan *et al.*, 2007), (Mármol, Girao *et al.*, 2010) and (Jøsang, Quattrociocchi *et al.*, 2011), some issues are still not taken into consideration, such as the credibility of referees, the update of the trust degree of the user at any intervention, the volatility of the rating and the feedback and the concordance between the given rating which is a scalar value and the textual feedback associated with it.
In contrast to those TRS, our proposed design treats these issues and uses an algorithm that includes semantic analysis of textual feedback in order to calculate the trust degree of the user and a most trustful reputation score for the product. In what follows, we will give an overview of the issues that are mostly neglected by related works and briefly explain the Reputation System proposed in their work.
Reputation systems are used to rate many objects, such as websites, services, products, customers, and also agents in interactive networks. Then trust reputation degree is calculated based on ratings given by many users using a specific aggregation method. For instance, the authors (Abdel-Hafez, Xu *et al.*, 2014) propose a model based on a weighted average method, where the weights are generated using the normal distribution. Their algorithm gives interesting results on either sparse or dense datasets.
Indeed, many methods use weighted average as an aggregator for the ratings, where the weight can represent the user's reputation, the time when the rating was given, or the distance between the current reputation score and the received rating (Jøsang, Quattrociocchi *et al.*, 2011).

3.1. Related work and the relying party's credibility issue

(Jøsang, Hayward *et al.*, 2008) propose a method that uses subjective logic in order to analyse trust network (TNA-SL). This approach aims to model in a simple way the relationship between different agents. A single arc means a single trust relationship between two nodes A and B [A; B] meaning that A trusts B. The authors performed a filtering method based on the principle of finding the most reliable paths or interactions between agents.
In e-commerce, we do not only need to generate trustful scores for product, users and feedback but we also need to propagate them through a trust reputation network. In the literature, a reputation network is defined as a network that can be represented by a graph with its nodes related to each other by an arc, multiple arcs or none. The graph reflects the trust network. The nodes represent users or agents interacting with each other directly or indirectly by giving recommendations and rating products. Arcs can refer to the information flow between agents when propagating ratings and recommendation from one to another.
However, trust has levels that are mentioned by ratings or textual feedback… etc. As a result, each arc must have its trust score. This trust relation is set up thanks to a transaction or a recommendation but obviously through something tangible like an evidence of trust. In fact, if A trusts B that means B has a positive reputation whether or not he deserves it. Because somehow we would have a group of agents knowing and trusting each other in the network in order to be seen as trustful by others and then falsely rate other agents or products. Therefore, each user should be analysed alone.
Consequently, we should take into consideration the trust degree of the arc and also the trust degree of the nodes, because the arc could be feedback given by B and used by A to deal with a transaction. That's why A trusts B.

In our approach, for each user who wants to leave a rating (appreciation) and a textual feedback (semantic feedback), we analyse his intervention using our algorithm. Indeed after verification, the user's recommendation is going to be accessible by any other user and then it is a recommendation for everyone interested or not in the product. Then, we assume that we have a path relaying every user. What is most important is to analyse at any intervention the user's attitude in order to deduce the user's intention concerning the rating of that specific product.

3.2. Related work and the trust update issue

Another factor, which is important in the analysis realized by a TRS, is the date of the creation or the establishment of the arc. The most recent arc, which relays two nodes having the same interest in a topic or a product, is more meaningful and useful than an old one. We can add an attribute «Date» in the graph table in the database. Date represents the date of the creation of the arc relating A to B on product.
Time is important in calculating a reputation score. Using time, among other features, to calculate trust scores, old ratings should get less weight than current ones using a specific method related to the age of the rating in time (Abdel-Hafez, Xu *et al.*, 2014). On the other hand, the age of rating issue, the trust update is a very important key contributing to the robustness of a TRS (Jøsang and Golbeck, 2009).
In fact, the authors (Ghazizadeh, Ehsaei *et al.*, 2012) think that the transitivity of trust is a derived trust from an existing trust between agents.
For example, we may find that Alice who trusts Bob doesn't trust Charlie because she has another friend who is Ben and she trusts him more than Bob. Ben doesn't trust Charlie so Alice will not trust Charlie. This arc is multiple and has two different statements and results with the same scope. And Bob may even change his mind after a bad transaction or situation with Charlie. In this case, we have to pay attention to the update of statements or feedback taken into consideration in the calculus of reputation score.
In our algorithm, at every intervention of a user, we update his already calculated trust degree if he has already participated in rating and commenting on a product. If not, we generate for him a new trust degree, according to his attitude modelled by «liking» or «disliking» prefabricated feedback concerning the same product displayed to him, in order to validate his own rating and feedback (see sect. 4). The choice of the prefabricated feedback is done on purpose because they are fabricated based on the product and its features and semantically analysed by text mining tools to generate their trust score. They are more adequate and specific than user's feedback in a raw state. Moreover, the system proposes to the user different feedback related to the same product but with different types (positive, negative, mitigated), at least three types of feedback, and he or she can have more: it is up to them.

3.3. Related work and the technique of reward and punishment in TRS

In fact, one of the biggest problem for reputation systems is users who falsely give ratings using many "faking identities" or none. In our TRS, even though the user has given a rating, we allow him to give ratings as much as he wants to. However, if at any intervention he changes his identity we consider him as a new user and we calculate a new trust degree which plays the role of a coefficient according to his rating. A possible example for such a rating method might be school marks and coefficient (see sect. 4). To demonstrate the impact of a mark, the coefficient must be higher and vice versa because it is an arithmetic operation (a multiplication). If the user is trustful then his trust degree will be higher and will considerably impact the global rating of the product.
(Liu, Zhang *et al.* 2011) proposes a novel Reputation System based on dynamic coalition formation, where buyers with similar subjectivity and rich experience will be awarded virtual credits for helping others find trustworthy sellers to conduct business successfully.
Additionally, the authors (Jøsang, Quattrociocchi *et al.*, 2011) use an approach that aims to calculate the trust weight. Once the transaction is carried out between the Web Service Providers WSP and the Web Service Consumers WSC, a reward or punishment is assigned to users and WSPs according to the accuracy and reliability of their recommendations (Jøsang, Quattrociocchi *et al.*, 2011). They mainly focus on the punishment and reward of users (it is equal for WSPs). So WSP1 sends the satisfaction of the user who asked for the service. A simple mechanism will be established to measure the divergence between the final satisfaction of the user and the previously given recommendation of users.
Yet, those recommendations can be untrustworthy. To overcome this problematic, our approach aims to reward users virtual credit which represents their credibility (trust degree) if they like a trustworthy feedback or unlike an untrustworthy one. And we punish them if they like an untrustworthy feedback or unlike a trustworthy one. However, the reward and the punishment have levels and degrees depending on the degree of the trustworthiness of the feedback (see sect.4.2). Furthermore, the feedback trustworthiness is determined by a semantic analysis system that aims to generate and determine the sentiment orientation of the feedback or the user's review. This sentiment orientation system should contain a semi-supervised machine learning to help the system get the right domain ontology according to each product and generate consequently a reliable sentiment orientation for each feedback.

3.4. State of the art of text mining algorithms

In this state of the art, we will focus on algorithms that analyse text in order to extract opinion and sentiment. This analysis can be called opinion mining. In the literature, research on opinion mining basically started with using a specific lexicon composed of opinion and sentiment words, for instance, great, nice, the best, bad, poor, like, dislike, etc. which have different types and forms (verbs, adjectives, nouns, etc.). Many researchers such as (Lin, Wilson *et al.*, 2006), (Goldberg and Zhu, 2006) have worked on mining these opinion words in order to identify their semantic orientations (positive, negative or neutral). However, gathering these words and storing them in a database is not sufficient. Linguistic and semantic rules must be defined and used to detect the sentiment generated by the use of these words in a specific context. The authors (Ding, Liu *et al.*, 2008), (Kanayama and Nasukawa, 2006) have worked and improved methods of identifying linguistic/semantic rules that identify opinion words and exploit them in the generation of sentiment orientations. In fact, gathering opinion words in a thesaurus or in corpus like (Kim and Hovy, 2004) is clearly inconsistent since there are always new reviews on new products written in different languages by different users, etc. When users' context, products and languages change opinion words and products' features may change also. In fact, product review classification techniques can be classified into two main approaches, those based on lexical resources and neutral language processing such as (Lin, Wilson *et al.*, 2006) and (Goldberg and Zhu, 2006) and those using machine learning algorithms such as (Hamouda and Rohaim, 2010). In this paper, the approach presents the results of applying the SentiWordNet lexical resource of opinion words which employs different techniques to the problem of automatic sentiment classification of reviews.

3. Related work

Many authors such as (Liu, Zhang *et al.*, 2011), (Steinbrecher, Groß *et al.*, 2009), (Josang, Roslan *et al.*, 2007), (Golbeck and Hendler, 2004), (Ghazizadeh and Hussin, 2012) propose in their work several TRS architectures with different algorithms to calculate the reputation score related to the product. Nevertheless, few academic works on TRS have been devoted to the inclusion of the semantic analysis of feedback in the calculus of the trust score of the product and especially the trust degree of the user. Even in studies attempting to provide more complex reputation methods such as (Josang, Roslan *et al.*, 2007), (Mármol, Girao *et al.*, 2010) and (Jøsang, Quattrociocchi *et al.*, 2011), some issues are still not taken into consideration, such as the credibility of referees, the update of the trust degree of the user at any intervention, the volatility of the rating and the feedback and the concordance between the given rating which is a scalar value and the textual feedback associated with it.
In contrast to those TRS, our proposed design treats these issues and uses an algorithm that includes semantic analysis of textual feedback in order to calculate the trust degree of the user and a most trustful reputation score for the product. In what follows, we will give an overview of the issues that are mostly neglected by related works and briefly explain the Reputation System proposed in their work.
Reputation systems are used to rate many objects, such as websites, services, products, customers, and also agents in interactive networks. Then trust reputation degree is calculated based on ratings given by many users using a specific aggregation method. For instance, the authors (Abdel-Hafez, Xu *et al.*, 2014) propose a model based on a weighted average method, where the weights are generated using the normal distribution. Their algorithm gives interesting results on either sparse or dense datasets.
Indeed, many methods use weighted average as an aggregator for the ratings, where the weight can represent the user's reputation, the time when the rating was given, or the distance between the current reputation score and the received rating (Jøsang, Quattrociocchi *et al.*, 2011).

3.1. Related work and the relying party's credibility issue

(Jøsang, Hayward *et al.*, 2008) propose a method that uses subjective logic in order to analyse trust network (TNA-SL). This approach aims to model in a simple way the relationship between different agents. A single arc means a single trust relationship between two nodes A and B [A; B] meaning that A trusts B. The authors performed a filtering method based on the principle of finding the most reliable paths or interactions between agents.
In e-commerce, we do not only need to generate trustful scores for product, users and feedback but we also need to propagate them through a trust reputation network. In the literature, a reputation network is defined as a network that can be represented by a graph with its nodes related to each other by an arc, multiple arcs or none. The graph reflects the trust network. The nodes represent users or agents interacting with each other directly or indirectly by giving recommendations and rating products. Arcs can refer to the information flow between agents when propagating ratings and recommendation from one to another.
However, trust has levels that are mentioned by ratings or textual feedback… etc. As a result, each arc must have its trust score. This trust relation is set up thanks to a transaction or a recommendation but obviously through

something tangible like an evidence of trust. In fact, if A trusts B that means B has a positive reputation whether or not he deserves it. Because somehow we would have a group of agents knowing and trusting each other in the network in order to be seen as trustful by others and then falsely rate other agents or products. Therefore, each user should be analysed alone.
Consequently, we should take into consideration the trust degree of the arc and also the trust degree of the nodes, because the arc could be feedback given by B and used by A to deal with a transaction. That's why A trusts B.
In our approach, for each user who wants to leave a rating (appreciation) and a textual feedback (semantic feedback), we analyse his intervention using our algorithm. Indeed after verification, the user's recommendation is going to be accessible by any other user and then it is a recommendation for everyone interested or not in the product. Then, we assume that we have a path relaying every user. What is most important is to analyse at any intervention the user's attitude in order to deduce the user's intention concerning the rating of that specific product.

3.2. Related work and the trust update issue

Another factor, which is important in the analysis realized by a TRS, is the date of the creation or the establishment of the arc. The most recent arc, which relays two nodes having the same interest in a topic or a product, is more meaningful and useful than an old one. We can add an attribute «Date» in the graph table in the database. Date represents the date of the creation of the arc relating A to B on product.
Time is important in calculating a reputation score. Using time, among other features, to calculate trust scores, old ratings should get less weight than current ones using a specific method related to the age of the rating in time (Abdel-Hafez, Xu *et al.*, 2014). On the other hand, the age of rating issue, the trust update is a very important key contributing to the robustness of a TRS (Jøsang and Golbeck, 2009).
In fact, the authors (Ghazizadeh, Ehsaei *et al.*, 2012) think that the transitivity of trust is a derived trust from an existing trust between agents.
For example, we may find that Alice who trusts Bob doesn't trust Charlie because she has another friend who is Ben and she trusts him more than Bob. Ben doesn't trust Charlie so Alice will not trust Charlie. This arc is multiple and has two different statements and results with the same scope. And Bob may even change his mind after a bad transaction or situation with Charlie. In this case, we have to pay attention to the update of statements or feedback taken into consideration in the calculus of reputation score.
In our algorithm, at every intervention of a user, we update his already calculated trust degree if he has already participated in rating and commenting on a product. If not, we generate for him a new trust degree, according to his attitude modelled by «liking» or «disliking» prefabricated feedback concerning the same product displayed to him, in order to validate his own rating and feedback (see sect. 4). The choice of the prefabricated feedback is done on purpose because they are fabricated based on the product and its features and semantically analysed by text mining tools to generate their trust score. They are more adequate and specific than user's feedback in a raw state. Moreover, the system proposes to the user different feedback related to the same product but with different types (positive, negative, mitigated), at least three types of feedback, and he or she can have more: it is up to them.

3.3. Related work and the technique of reward and punishment in TRS

In fact, one of the biggest problem for reputation systems is users who falsely give ratings using many «faking identities» or none. In our TRS, even though the user has given a rating, we allow him to give ratings as much as he wants to. However, if at any intervention he changes his identity we consider him as a new user and we calculate a new trust degree which plays the role of a coefficient according to his rating. A possible example for such a rating method might be school marks and coefficient (see sect. 4). To demonstrate the impact of a mark, the coefficient must be higher and vice versa because it is an arithmetic operation (a multiplication). If the user is trustful then his trust degree will be higher and will considerably impact the global rating of the product.
(Liu, Zhang *et al.* 2011) proposes a novel Reputation System based on dynamic coalition formation, where buyers with similar subjectivity and rich experience will be awarded virtual credits for helping others find trustworthy sellers to conduct business successfully.
Additionally, the authors (Jøsang, Quattrociocchi *et al.*, 2011) use an approach that aims to calculate the trust weight. Once the transaction is carried out between the Web Service Providers WSP and the Web Service Consumers WSC, a reward or punishment is assigned to users and WSPs according to the accuracy and reliability of their recommendations (Jøsang, Quattrociocchi *et al.*, 2011). They mainly focus on the punishment and reward of users (it is equal for WSPs). So WSP1 sends the satisfaction of the user who asked for the service. A simple mechanism will be established to measure the divergence between the final satisfaction of the user and the previously given recommendation of users.
Yet, those recommendations can be untrustworthy. To overcome this problematic, our approach aims to reward users virtual credit which represents their credibility (trust degree) if they like a trustworthy feedback or unlike

an untrustworthy one. And we punish them if they like an untrustworthy feedback or unlike a trustworthy one. However, the reward and the punishment have levels and degrees depending on the degree of the trustworthiness of the feedback (see sect.4.2). Furthermore, the feedback trustworthiness is determined by a semantic analysis system that aims to generate and determine the sentiment orientation of the feedback or the user's review. This sentiment orientation system should contain a semi-supervised machine learning to help the system get the right domain ontology according to each product and generate consequently a reliable sentiment orientation for each feedback.

3.4. State of the art of text mining algorithms

In this state of the art, we will focus on algorithms that analyse text in order to extract opinion and sentiment. This analysis can be called opinion mining. In the literature, research on opinion mining basically started with using a specific lexicon composed of opinion and sentiment words, for instance, great, nice, the best, bad, poor, like, dislike, etc. which have different types and forms (verbs, adjectives, nouns, etc.). Many researchers such as (Lin, Wilson *et al.*, 2006), (Goldberg and Zhu, 2006) have worked on mining these opinion words in order to identify their semantic orientations (positive, negative or neutral). However, gathering these words and storing them in a database is not sufficient. Linguistic and semantic rules must be defined and used to detect the sentiment generated by the use of these words in a specific context. The authors (Ding, Liu *et al.*, 2008), (Kanayama and Nasukawa, 2006) have worked and improved methods of identifying linguistic/semantic rules that identify opinion words and exploit them in the generation of sentiment orientations. In fact, gathering opinion words in a thesaurus or in corpus like (Kim and Hovy, 2004) is clearly inconsistent since there are always new reviews on new products written in different languages by different users, etc. When users' context, products and languages change opinion words and products' features may change also. In fact, product review classification techniques can be classified into two main approaches, those based on lexical resources and neutral language processing such as (Lin, Wilson *et al.*, 2006) and (Goldberg and Zhu, 2006) and those using machine learning algorithms such as (Hamouda and Rohaim, 2010). In this paper, the approach presents the results of applying the SentiWordNet lexical resource of opinion words which employs different techniques to the problem of automatic sentiment classification of reviews.

4. Our Trust Reputation System design

4.1. Algorithm Description

The user starts by giving an appreciation (rating) and textual feedback about a specific product. When he clicks on submit in order to validate the given information, we're going to redirect the user to another interface showing this message for example: «Please give us your opinion about the following feedback before validating the information you gave below:»

In fact, in this interface we will find different types of feedback from the database. This feedback can be fabricated in order to summarize numerous users' feedback stored in the database. The generated feedback can be stored in another knowledge base. As much as we add feedback to the ordinary data base, we will fill the knowledge database with prefabricated feedback using text mining algorithms and tools. However, some users may give already summarized feedback that can be directly included in the knowledge database. Indeed, there are many text mining and data mining algorithms and tools that could search for the most appropriate feedback which are, first of all, related to the product and that can recapitulate and summarize most of each type of the users' feedback.

Actually, before sending the user's feedback and appreciation about the product to the trust reputation system, we have to verify the agreement between them in order to avoid and eliminate contradiction or malicious programs attacking our system. In the redirected interface, we will display different types of feedback. The user can specify the number of feedback to be liked or disliked. Of course, we can also specify the minimum and the maximum number of feedback to be displayed by the user.

In fact, through this redirection, we are trying to detect and analyse the user's intention through his intervention on the e-commerce application. Hence, we examine and evaluate his intention using other pre-fabricated feedback with different types. Of course, we already have the trustworthiness of each feedback. Consequently, we use our reputation algorithm studied in section [4.2] in order to generate the user trust degree which plays the role of a coefficient and then rectify his appreciation according to his trust degree and generates the score of the feedback. Indeed, each feedback has trustworthiness in a [-5, 5] range. The closer the trustworthiness is to 5, the more trustworthy the feedback is. The closer the trustworthiness is to -5, the less trustworthy the feedback is.

If the feedback is trustworthy, its score would be in]0,5] or else it would be in [-5,0].

4.2. TRS algorithm

The reputation algorithm used in this TRS uses a semantic feedback analysis in order to generate a trustful reputation score for the product.
Actually, we have three types of feedback:

- **Positive feedback**: represents opinions that express a positive point of view about the product. Those ameliorative opinions contain a positive content concerning the product. Then, the positive adjective refers to the nature of feedback not to its trustworthiness.
- However, each feedback, regardless of its type, can have either a positive trustworthiness or a negative trustworthiness. Whether it is a positive trustworthiness or a negative one, it is gradual: it has degrees fluctuate within a level.
- **Negative feedback**: represents opinions talking negatively about the product. Logically, the users giving such opinions are not satisfied with the product in question. This feedback could be telling the truth, part of the truth or could be far from the truth. That's why each feedback has its trustworthiness represented by a fluctuation between -5 and 5.
- **Mitigated feedback**: represents feedback that describes positively some product's features and negatively describes others. Its trustworthiness is in the threshold [-5.5].
- **Contradictory feedback**: represents feedback with a contradictory content. For example, a feedback where the user is not talking about the specified product but another one such as he/she claims that the camera of a mobile phone is great and later in the same opinion says that the camera is very bad. Subsequently, we have to start by detecting the contradictory feedback. As a result, we need semantic analysis system that is able to detect the contradiction in a specific content related to a product. We have to personalize the semantic analysis for each category of product in order to have a TRS working on various domains and products. For instance, if the user says that "the swimming pool of the hotel - which can't offer one - is not clean", the algorithm must be able to detect this discrepancy. For each product, we give the property of the algorithm as an input; if there is no similarity we can consider it as a contradiction, but the agreement has to include the meaning. If the user writes that the negative thing about this hotel is that there is no swimming pool and he's telling the truth then obviously the presence of an absent feature of the product in a feedback doesn't lead to a contradiction.

In this paper, we will not discuss either the text mining algorithm or the sentiment analysis system. We're going to develop it in future work.
Before sending the user's feedback and appreciation of the product to the trust reputation system, we have to verify the concordance or the alliance between them in order to avoid any contradiction.

Pseudo-code to verify the concordance between the rating and the textual feedback:

```
Boolean concordance;
concordance =Test_ concordance (int appreciation, string feedback) ;
If (concordance)
URL (url_feedbacks_interface); //redirection to the feedbacks interface
Else
URL (url_page); // we thank the user for his intervention and we put him temporally in a
//blacklist for unconformity
```

figure 1. Verification of the agreement between the rating and the textual feedback

After verifying the coherence or the match between the appreciation and the textual feedback, we redirect the user to a selection of prefabricated feedback (Rahimi and El Bakkali, 2012). Then the user clicks on 'like' or 'dislike' according to each type of feedback. The click will be managed to provide important data to generate the trust degree of the user. The function uses the feedback id as a parameter to get its trustworthiness from the knowledge base. We also need to get the previous trust degree of the user if he has already been engaged in a transaction or he has used the application for rating purposes. The user's choice of either 'like' or 'dislike' is an important parameter to determine his trustworthiness.

Pseudo-code for a function that gets parameters used in the calculus of the trust degree of the user:

```
Function get-infos-click (int id feedback) as list{
double Feedtrustworth;//the feedback trustworthiness stored in knowledge base
// its value is between -5 and 5
String Userchoice;// represents the user's choice either it is a "like" or a "dislike"
double degree_trust_user; //a double value representing the trust degree of the user [-5;5]
double scorefeed; //a double value between -5 and 5
degree_trust_user=get_trust_degree_user (login);
Feedtrustworth= getfeedtrustworth (idfeedback); // this function gets the trustworthiness of the feedback either positive or negative value between -5 and 5
Userchoice=getuserchoice (idfeedback); // this function get the user choice after the click //from the interface
List listinfos=[ Feedtrustworth, Userchoice, degree_trust_user];
        Return listinfos;}
```

figure 2. Getting information to calculate the trust

After extracting the parameters going to be used in the next calculus, we will calculate the trust degree of the user taking into consideration the type of the feedback's trustworthiness and the user's choice. The calculus of the trust degree of the user can be an update if the user already has a trust degree. As seen in the function below, «get_trust_degree_user (login)» uses the login of the user in order to select his last trust degree. We can add as a parameter the id of the product in order to select his trust degree for a specific product. Consequently, we select the trust degree of the user, who logged in with a specific «login» and calculate his intervention rating on a specific product (id_product). In that case, we have to generate a global trust degree for the user by product and a global trust degree according to his general interventions on the e-commerce platform. This method could be developed in a future paper.

Pseudo-code for the calculus of the trust degree of the user:

```
        function calculate_degree_trust_user () as double{
        list listinfos;
        Int idfeedback=get_idfeedback();
        Listinfos=get_infos_click (idfeedback);
        List listScore;
        Feedtrustworth= Listinfos[0];
        Userchoice= listinfos [1]

        Degree_trust_user=Listinfos[2];

        if (0<feedtrustworth<=1.5) and (userchoice="like")
        Or (-1.5=<feedtrustworth<=0) and (userchoice="dislike")
                Degree_trust_user+=0.25

        If (1.5<feedtrustworth<=2.5) and (userchoice="like")
        Or(-2.5=<feedtrustworth<-1.5) and (userchoice="dislike")
                Degree_trust_user+=0.5

        If (2.5<feedtrustworth<=3.5) and (userchoice="like")
        Or(-3.5=<feedtrustworth<-2.5) and (userchoice="dislike")
                Degree_trust_user+=0.75

        If (3.5<feedtrustworth<=5) and (userchoice="like")
        Or (-5=<feedtrustworth<-3.5) and (userchoice="dislike")
                Degree_trust_user+=1
```

figure 3. The calculus of the trust degree

Our proposed algorithm rewards the user by incrementing his trust degree if he likes a trustworthy feedback or he dislikes an untrustworthy one. The incrementing value (reward) depends on the value of the feedback trustworthiness. When the user's choice is a «like», the greater the feedback trustworthiness is, the greater the reward would be and vice versa. And when the user dislikes a feedback, the greater the untrustworthiness of the feedback is, the greater the reward would be and vice versa.
At this point, it is essential to respect the threshold [-5; 5]. We can call it the values normalization or values standardization. In future work, we can give a detailed method of values normalization in order to be more concise. All the trust scores must be in a [-5; 5] range. For example, if the score is -7, the reputation algorithm approaches -7 to -5 to respect the threshold [-5; 5]. However, this approach is less precise than to multiply the score by a coefficient or to have a larger threshold such as [-10; 10].

Pseudo-code for the values standardization:

```
// to respect the threshold [-5;5]
If (Degree_trust_user<-5)
    Degree_trust_user=-5;
Else if (Degree_trust_user>5)
    Degree_trust_user=5;
Return degree_trust_user;
}
// the end of the function
```

figure 4. The scores' standardization

In fact, the function returns the trust degree of the user updated according to his current participation. As a result, if his trust degree is positive we will take into consideration his rating realized in the first interface of the e-commerce application before redirection. However, if his trust degree is negative, we will not include his appreciation in the calculus of the global trust score of the product and we can preserve his feedback in order to use it to generate other feedback. As a result, his feedback would be considered as untrustworthy as provider and vice versa.
Afterwards, we have to generate the global trust reputation score of the product using the user's appreciation (rating) and his trust degree. In fact, a possible example for such a rating method might be school marks and coefficients. Actually, at school, when a course is important for a certain field, its coefficient would be great and then the effect of its mark would be greater. In the same context, we consider the trust degree of the user as a coefficient and his appreciation as a mark. Consequently, to calculate the global trust score of the product, we add all the appreciation values multiplied by their respective coefficient and then divide the result of the total by the sum of all coefficients:

$$\frac{X+bY}{a+b}=\frac{\sum_{1st\ user}^{last\ user} \text{Appreciation} * \text{trust degree}}{\sum_{1st\ user}^{last\ user} \text{trust degree}}$$
$$=\frac{\text{the summation of all users' appreciations until the new user} * \text{their trust degree respectively}}{\text{the summation of all users' trust degree}}$$

figure 5. The calculus of the trust degree of the product

- *«X»* represents the summation of all users' appreciations.
- *«Y»* represents the new appreciation given by the user.
- *«b»* represents the new coefficient to be added, and "a" represents the summation of all users' trust degrees.

We can store the *«X»* and the *«a»* in different areas so we can get them separately and then calculate easily:

$$\frac{X + bY}{a + b}$$

4. Conclusion

Lack of trust is always considered as an obstacle to users dealing with electronic transactions. Trust Reputation Systems aim at creating trust and propagating it in online communities, while giving results one we can act on. These results such as trust reputation scores help users make the right decision about purchasing a product or not. However, users may provide false ratings and falsified semantic feedback which might be both the base of the calculus of the reputation score.

To overcome this limitation, we have proposed a TRS design that uses both user's ratings and feedback in order to calculate trust reputation scores of users and products. This approach is based on an intelligent layer that semantically analyses the user's feedback to determine its sentiment about the product and its trustworthiness. Additionally, a Sentiment Orientation System helps fabricate reviews summarising users' feedback. Besides, the Reputation Algorithm analyses the user's attitude toward the prefabricated reviews in order to calculate the trust degree of both the user and the product using a coefficient-average method. In future work, we aim to develop the Sentiment Orientation System and implement it with the Reputation Algorithm in order to obtain a simulation of the TRS and to evaluate its effectiveness in an e-commerce platform.

5. References

Abdel-Hafez, A, Xu, Y, and Josang, A (2014): A Normal-Distribution Based Reputation Model. *11th International Conference on Trust, Privacy & Security in Digital Business (TrustBus 2014) (to appear)*. Munich, September 2014.

Ding, X, Liu, B. and Yu, P (2008). A Holistic Lexicon-Based Approach to Opinion Mining. *Proceedings of the first ACM International Conference on Web search and Data Mining (WSDM'08)*.

Ehsaei, F., Hussin, A (2012) Acceptance of Feedback in Reputation Systems: The Role of Online Social Interactions Information Management and Business Review Vol. 4, No. 7, pp. 391-401, (ISSN 2220-3796).

Golbeck, J., Hendler, J., (2004) Inferring Reputation on the Semantic Web. *In the proceedings of WWW 2004,* New York, NY USA. ACM.

Goldberg, A.B. and Zhu, J (2006). Seeing stars when there aren't many stars: Graph-based Semi-supervised Learning for Sentiment Categorization. *HLT-NAACL 2006 Workshop on Textgraphs: Graph-based Algorithms for Natural Language Processing.*

Gutowska, A, Sloane, A (2009) Modelling the B2C Marketplace: Evaluation of a Reputation Metric for e-commerce. *Proceedings of Web Information Systems and Technologies - WEBIST*, pp. 212-226

Hamouda, A., Rohaim, M. (2010) Reviews Classification Using SentiWordNet Lexicon.*The Online Journal on Computer Science and Information Technology (OJCSIT)* Vol. (2) – No. (1) pp:120-123.

Jøsang, A., Golbeck, J., (2009) Challenges for Robust Trust and Reputation Systems. *In the proceedings of the 5th International Workshop on Security and Trust Management (STM 2009),* Saint Malo, France, September.

Jøsang, A., Hayward, R., Pope, S (2008) Trust Network Analysis with Subjective Logic. *Proceedings of the Second International Conference on Emerging Security Information, Systems and Technologies (SECURWARE 2008),* Cap Esterel, France.

Jøsang, A., Quattrociocchi, W., Karabeg, D., (2011) Taste and Trust. *In the Proceedings of the IfiP internationa conference of trust management IfiPTM 2011,* Copenhagen, June/July 2011.

Josang, A., Roslan, I., and Colin, B., (2007) A survey of trust and reputation systems for online service provision. *In the proceedings of Decision Support Systems 43(2)*:pp. 618-644.

Kanayama, H. and Nasukawa, T (2006). Fully Automatic Lexicon Expansion for Domain-Oriented Sentiment Analysis. *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP'06).*

Kim, S. and Hovy, E (2004). Determining the Sentiment of Opinions. *Proceedings of the 20th International Conference on Computational Linguistics (COLING'04).*

Komiak, S., (2010) The Effects of Perceived Information Quality and Perceived System Quality on Trust and Adoption of Online Reputation Systems. *In the Proceedings of AMCIS 2010.*

Lin, W.H., Wilson, T., Wiebe, J. and Hauptman, A (2006). Which Side are You On? Identifying Perspectives at the Document and Sentence Levels. *Proceedings of the Tenth Conference on Natural Language Learning CoNLL'06.*

Liu, Y., Zhang,J., Zhu, Q., (2011) Design of a Reputation Systembased on Dynamic Coalition Formation. *Proceedings of Third International Conference, SocInfo 2011,* Singapore, pp 135-144.

Mármol, F., Girao, J., Pérez, G., (2010) TRIMS, a privacy-aware trust and reputation model for identity management systems. *In the proceeding of Computer Networks, 54(16)*:2899-2912.

Mayer, R., Davis J., Schoorman, D., (1995) An Integrative Model of Organizational Trust. *In the proceedings of The Academy of Management Review.*Vol. 20, No. 3 (Jul., 1995), pp. 709-734.

Rahimi, H., El Bakkali, H., (2012) Toward a New Design of Trust Reputation Systemin e-commerce,. *In the proceedings of ICMCS (International conference on Multimedia computing and systems,* Tangier, Morocco, IEEE 2012).

Rahimi, H., El Bakkali, H., (2012) Etat de l'art de la gestion de la confiance et des Systèmes de réputation en e-commerce. *In the proceedings of JNS2 (2ème journée nationale de la Sécurité,* ENSA, Marrakech, Morocco, IEEE explore 2012).

Steinbrecher, S., Groß, S., Meichau, M., (2009) Jason: A Scalable Reputation Systemfor the semantic Web. *In the proceedings of IfiP International Federation for Information Processing.*

Zhao, H., Yang, X., Li:, X (2010) WIM: A Wage-based Incentive Mechanism for Reinforcing Truthful Feedback in Reputation Systems. *In the Proceedings of the Global Communications Conference, GLOBECOM 2010*, 6-10 December 2010, Miami, florida, USA. IEEE 2010.