Spam Detection Approach for Cloud Service Reviews Based on Probabilistic Ontology

Emna Ben-Abdallah(✉), Khouloud Boukadi(✉), and Mohamed Hammami

Mir@cl Laboratory, Sfax University, Sfax, Tunisia
emnabenabdallah@ymail.com, khouloud.boukadi@gmail.com

Abstract. Online reviews provide a vision on the strengths and weakness of products/services, influencing potential customers' purchasing decisions. The fact that anybody can leave a review provides the opportunity for spammers to write spam reviews about products and services for different intents. To counter this problem, a number of approaches for detecting spam reviews have been proposed. However, to date, most of these approaches depend on rich/complete information about items/reviewers, which is not the case of Social Media Platforms (SMPs). In this paper, we consider well known spam features taken from the literature to them we add two new ones: the user profile authenticity to allow the detection of spam review from any SMP and opinion deviation to verify the opinion truthfulness. To define a common model for different SMPs and to cope with the incompleteness of information and uncertainty in spam judgment, we propose a Review Spam Probabilistic Ontology (RSPO) based approach. Probabilistic Ontology is defined using Probabilistic Web Ontology Language (PR-OWL) and the probability distributions of the review spamicity is defined automatically using a learning approach. The herein reported experimental results proved the effectiveness and the performance of the approach.

Keywords: Spam review detection · Probabilistic ontology
Social media

1 Introduction

Nowadays, online reviews are an important source of information for consumers to evaluate online services and products before deciding which product and which provider to choose. In fact, they have a significant power to influence consumers' purchasing decisions. Through social network sites (SNS) such as Facebook, which are considered as the most used one according to the statistics presented in Pew 2018 [7], consumers can freely give feedback, exhibit their reactions to a post or product, share their opinion with their peers and also share their grievances with the companies. However, SNSs cannot yet detect spam reviews and even fake profiles in-time, and hence discriminating between real

H. Panetto et al. (Eds.): OTM 2018 Conferences, LNCS 11229, pp. 534–551, 2018.
https://doi.org/10.1007/978-3-030-02610-3_30

and fake profiles is difficult for non-technically savvy users. Being aware of this, an increasing number of companies have organized spammer review campaigns, in order to promote their products and gain an advantage over their competitors by manipulating and misleading consumers. Hence, this makes trust arise as a crucial factor on the web.

Research on this topic has cast the problem of spam review and spammer user detection into a binary classification: a review is either credible or spam and a user is either honest or spammer. To this end, spam feature clues (behavioral and linguistic features) are defined to identify the spam reviews and spammer users. These features are determined from meta-information (date of review, rate, history of the user, etc.) and from review text. Behavioral features are mostly geared from platform review sites such as Yielp and Amazon where the meta-information about the user's history are almost available. Contrariwise, this is not always the case of SNSs like Facebook. Several existing studies [15,27] consider the review text for tackling spam reviews by using linguistic features such as, the average content and maximum content similarity; however such features are not considered to analyze the spamicity of the opinion. We believe that it is important to analyze the opinion for the spam review detection. In other words, spammer generally does not give the right opinion to defame or to promote a product/service.

To present the features, many approaches relied on graph/network based methods [26,27]. However, they do not pay attention to the concepts heterogeneity, for example the "profile" concept in Facebook is the same as "account" concept in review sites, also in review sites the "review" concept is similar to the "feedback" concept in Facebook. Since the social media environment is open, distributed, and semantically enabled, it is not only necessary to have spam detection techniques but also to empower these techniques with semantics to facilitate the quality access and the retrieval of credible reviews from any social media plateform. Besides, the spam judgments are subjective and uncertain in nature. In fact, we cannot affirm the clue of spamicity, or we cannot affirm that the review is spam or the reviewer is a spammer if it/he has spam features. For instance, one reviewer may use a fake profile to hide his identity but he writes a credible review, and vice versa. Moreover, if a review has some features depicting that it is a spam review, while others indicate that it is a credible one; thus leading to a confusing situation. Therefore, an approach that aims at resolving the heterogeneity problem of reviews description and reviewers of social media platforms and supporting the uncertainty of the spam review assessment is of paramount importance. This paper focus on how to reveal spammers and spam reviews from any social media platforms. Moreover, it sheds light on how inferring spam firstly from incomplete and ambiguous information related to spam feature clues and secondly by supporting the uncertainty of the spam judgment.

To cope with the problems mentioned above, we propose to rely on ontology to resolve the heterogeneity problem of social media platforms. However, traditional ontology does not support the uncertainty reasoning [9]. The probabilistic ontology has the merit of supporting the uncertainty, which could be

used to asses the spamicity of reviews in SMPs. Besides, we rely on learning based method to generate the probability distribution of the review spamicity. The choice of a learning based method to predict the review spamicity can be explained by two reasons: First, if the probability distributions are defined manually by domain experts, this can decrease the spam review detection performance. In fact, experts can not predict all spammers and spam review behaviors. Second, spammers may take advantage of the design and update their review to deceive the detection process.

Our proposed approach introduces Review Spam Probabilistic Ontology (RSPO) which describes relevant concepts for the detection of spammer users and spam reviews from social media platforms with the aim of facilitating the retrieval of credible reviews and the detection of spam information. This probabilistic ontology is defined using PR-OWL [11] and infers the degree of spamicity of reviews based on MEBN-learning (Multi Entity Bayesian Network learning) method [25].

The rest of the paper is organized as follows. Section 2 aims to define the probabilistic ontology and the PR-OWL language. Section 3 presents spam features used in this paper as clues of spamicity. The proposed RSPO ontology is depicted in Sect. 4. Experimental evaluations are presented in Sect. 5. Section 6 discusses the related works before drawing some conclusions and discussing some future work in Sect. 7.

2 Background

This section presents a brief overview of the probabilistic ontology and the Uncertainty Modelling Process for the Semantic Web (UMP-SW) methodology which form the basis of our work. A probabilistic ontology is an explicit, formal knowledge representation that expresses knowledge about a domain of application. This encompasses: types of entities, properties, relationships, processes and events that happen with the entities, statistical regularities that characterize the domain, inconclusive, ambiguous, incomplete, unreliable, and dissonant knowledge, and uncertainty about all the above forms of knowledge. Probabilistic ontologies are used for the purpose of comprehensively describing knowledge about a domain and the uncertainty associated with that knowledge in a principled, structured, and sharable way [9]. This has given birth to a number of new languages such as: PR-OWL [11], OntoBayes [30] and BayesOWL [12]. In this paper, we rely on PR-OWL to represent the RSPO. Actually, PR-OWL not only provides a consistent representation of uncertain knowledge that can be reused by different probabilistic systems, but also allows applications to perform plausible reasoning with that knowledge, in an efficient way [9]. This can be explained by the fact that PR-OWL is based on Multi-Entity Bayesian Network (MEBN) logic. MEBN extends Bayesian Networks (BN) to achieve first-order expressive power. MEBN represents knowledge as a collection of MEBN Fragments (MFrags), which are organized into MEBN Theories (MTheories). An MFrag (see Fig. 4) contains random variables (RVs) and a fragment graph representing dependencies among these RVs. An MFrag represents a repeatable

pattern of knowledge that can be instantiated as many times as needed to form a BN addressing a specific situation called situation-specific Bayesian Networks (SSBN), and thus can be seen as a template for building and combining fragments of a Bayesian network. An MFrag can contain three kinds of nodes: context nodes which represent conditions under which the distribution defined in the MFrag is valid, input nodes which have their distributions defined elsewhere and condition the distributions defined in the MFrag, and resident nodes with their distributions defined in the MFrag. Each resident node has an associated class local distribution which defines its distribution as a function of the values of its parents, namely Local Probability Distribution (LPD). The RVs in an MFrag can depend on ordinary variables. We can substitute different domain entities for the ordinary variables to make instances of the RVs in the MFrag.

In order to model and implement PR-OWL ontologies, Carvalho et al. proposed the Uncertainty Modelling Process for the Semantic Web (UMP-SW) methodology [10]. This methodology is consistent with the Bayesian network modelling methodology [18] and includes three main steps: model the domain, populate its Knowledge Base (KB), and perform reasoning based on both the model and the KB. The modelling step consists of three major stages: requirements, analysis and design, and implementation. These stages are borrowed from the Unified Process (UP) with some modifications to fit the ontology modelling domain.

3 Spam Feature Description

To infer the degree of spamicity/credibility of a review and reviewer, this paper relies on spam features [15,23] that fall into the categories as follows:

1. Review-Behavioral (RB) based features: This type of feature is based on the review meta-information and not on the review text itself. The RB category encompasses two features:
   - Early Time Frame (ETF): Spammers often review early to inflict spam as the early reviews can greatly impact people's sentiment on a product/service [22].

$$v_{etf} = \begin{cases} 0 & (T_i - F_i) \notin [0, \delta] \\ 1 - \frac{T_i - F_i}{\delta} & (T_i - F_i) \in [0, \delta] \end{cases} \tag{1}$$

     Where $T_i - F_i$ denote the period between the r_i (review i) date and the first review date. $\delta = 7$ months is a threshold for denoting earliness. $etf(r_i)$ takes value 1 if v_{etf} is greater than 0.5 otherwise it takes value 0.
   - Rate Deviation (RD) [22]: Spammers attempt to promote or demote products/services, their ratings can deviate from the average ratings given by other reviewers. Rating deviation is thus a possible behavior demonstrated by a spammer. This feature attains the value of 1 if the rating deviation of a review exceeds some threshold β ($\beta = 0.63$).

$$rd_i = \begin{cases} 1 & \frac{rt_{ij} - avg_{e \in E_{*j}} r(e)}{4} > \beta \\ 0 & otherwise \end{cases} \tag{2}$$

Where rt_{ij} refers to the rating given by the reviewer i towards an item j.

2. Review-Linguistic (RL) based features: Features in this category are based on the review text. In this work, we use two main features in RL category:
   - Ratio of Exclamation Sentence containing '!' (RES) [19]: Spammer put '!' in their sentences as much as they can to increase impression on users and highlight their reviews among other ones.

$$res(r_i) = \begin{cases} 1 & contain'!' \\ 0 & otherwise \end{cases} \tag{3}$$

   - Number of the first Personal Pronoun (NPP) [19]: Studies show that spammers use second personal pronouns much more than first personal pronouns.

$$npp(r_i) = \begin{cases} 1 & true \\ 0 & false \end{cases} \tag{4}$$

3. User-Behavioral (UB) based features: Relate to each user and encapasses two main features:
   - Reviewing Burstiness (BST) [21]: Spammers, always write their spam reviews in short period of time for two reasons: first, because they want to impact readers and other users, and second because they are temporal users, they have to write as much as reviews they can in short time.

$$v_{bst} = \begin{cases} 0 & (L_i - F_i) \notin [0, \tau] \\ 1 - \frac{L_i - F_i}{\tau} & (L_i - F_i) \in [0, \tau] \end{cases} \tag{5}$$

     Where τ is the time window parameter representing a burst ($\tau = 28$ days). $L_i - F_i$ present the time interval between the first and the last reviews written by the user i (u_i). $bst(u_i)$ takes value 1 if v_{bst} is greater than 0.5 otherwise it takes value 0.
   - Negative Ratio (NR) [21]: Spammers tend to write reviews which defame businesses which are competitor with the ones they have contact with, this can be done with destructive reviews, or with rating those businesses with low score. Hence, ratio of their scores tend to be low.

$$nr(u_i) = \begin{cases} 1 & average\_rate\_of\_user_{u_i} \leq 2 \\ 0 & otherwise \end{cases} \tag{6}$$

4. User-Linguistic (UL) based features: These features, which are extracted from the users' language, show how the users are describing their feelings or opinions about what they have experienced as a customer of a business. We use this type of features to understand how a spammer communicates in terms of wording. The Average Content Similarity (ACS) is considered in this work since it is largely adopted in the litterature.
   - ACS [14]: As crafting a new review every time is time consuming, spammers are likely to copy reviews across similar products. It is thus useful to capture the content similarity of reviews (using cosine similarity) of the

same author. We choose the maximum similarity to capture the worst spamming behavior.

$$acs(u_i) = \begin{cases} 1 & u_i \quad has\_similar\_reviews \\ & 0 \quad otherwise \end{cases} \tag{7}$$

5. Profile Authenticity (PA) feature: Besides the features depicted above, we propose in this work a new feature, namely profile authenticity, to detect spammers. It is more likely that people who write spam reviews hide their identities, especially in social network sites where it is easy to create a fake account. To differentiate between fake profiles and authentic ones, we choose the four most famous profile elements: the profile picture (exist or not), the number of friends, his location and professional information. Considering professional information, it can be explained by the fact that the proposed approach will be applied to detect spam reviews of cloud services, which are generally used by enterprises and not by individual users. Hence, a spammer may hide his enterprise, his workplace as well as his job.
6. Opinion Deviation (OD) feature: The use of opinion deviation feature aims at detecting, first, the unusual reviews (for example, $< 3 < 3 < 3 < 3 < 3 < 3$; Great!!!!!!!!!!!!!!!!); second the without-feature-reviews (for example, in the field of cloud service, reviews that do not contain any service property, such as World's Best Service!!! Just < 3 You!!!); and third the divergent opinions compared to the majority of reviews.
   In fact, many approaches have been used to detect deviations among which we can mention, the clustering based approach which is the most commonly developed [17]. For this reason, we use the clustering technique to identify divergent opinions (see Fig. 1) by calculating the outliers for each object. This factor depends on the distance from the object to the centroid of the cluster to which the object belongs. The algorithm starts iteratively by first finding the object with the maximum distance d_{max} to the cluster centroid thus:
   $$d_{max} = max_i\{||x_i - C_i||\}, \quad i = 1, 2, ..., N \tag{8}$$
   Outlier factors o_i, for each object are then calculated. An outlier factor (deviation) value for each object x_i is calculated using the Eq. 9.
   $$o_i = \frac{||x_i - C_i||}{d_{max}} \tag{9}$$
   Where $||x_i - C_i||$ is the distance between each object x_i and its allocated cluster centroid C_i. d_{max} is the maximum distance of a certain object to the cluster centroid/center. After all iterations, each object will have an outlier factor value that represents the object's deviation degree. All outlier factor values of the dataset are normalized to the range [0, 1]. The outlier factor value is compared with a predefined threshold value T that lies between 0 and 1. An outlier factor with a greater value is more likely to be a deviation. The object for which $o_i > T$ is considered a deviation. In order to annotate data

for the clustering, we adopt our previous work [8]. In particular, aligned with the cloud service domain, each review is presented as a set of service properties sp_j associated with their sentiment scores (as depicted in Fig. 2). The sentiment score is computed using [8] which presents a normalized average of the reviewer's sentiments scores about a service property in each review (the score of each sentiment is extracted from SentiwordNet [13]).

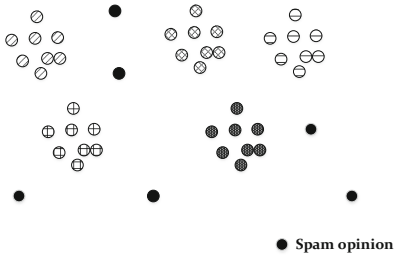

Fig. 1. Example of outlier objects

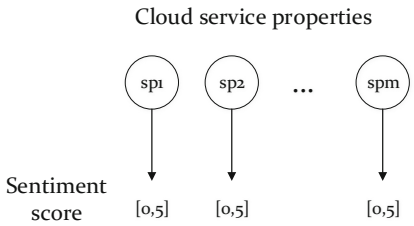

Fig. 2. Review form

4 Review Spam Probabilistic Ontology Modelling

After defining the features that will be used as clues to detect spammers and spam reviews, we should deal with the problem of modelization of these features and how to infer if the review is spam or not from the latter ones. The main challenges that hamper the review spam detection are: first, the subjectivity expectation of spamicity judgment, which makes the review spam inference uncertain and second, the incompleteness of spam features. This is can be explained by the fact that information about the user's history, review and profile is not always available on SMP. For this purpose, Review Spam Probabilistic Ontology (RSPO) is proposed in this work. The details of the RSPO modelling are presented in this section. The RSPO is created using the Uncertainty Model for the Semantic Web (UMP-SW) presented in Sect. 2. In particular, we deal with the RSPO modelling through three stages: Requirements, Analysis and Design, and Implementation.

4.1 Requirements

The main goal is to identify the likelihood of a particular review being spam. Requirement discipline draws out the goals, queries, and evidence for a particular system. To ensure the traceability of requirements, a specification tree is used. Each of the requirements is linked to its 'parent' requirement and every evidence is linked to its parent query, which in turn is linked to its higher-level goal. This arrangement helps trace the requirements.

Overall Goal of the RSPO is to determine either a review is credible or spam.

(1) Query: Does the reviewer have an authentic profile or a fake one?
   - Evidence: Look at the location information if it is available (on the reviewer's profile);
   - Evidence: Look at the enterprise information if it is available (on the reviewer's profile);
   - Evidence: Look at the job information if it is available (on the reviewer's profile);
   - Evidence: Look at the picture if it is available (on the reviewer's profile);
   - Evidence: Look at the friendship network number if it is greater than 50 (on the reviewer's profile);

(2) Query: Did the user write reviews to describe his experiences as a customer of a certain business?
   - Evidence: Look if the reviewer has similar reviews or not;

(3) Query: Has the reviewer a normal behaviour or suspicious one?
   - Evidence: Look at the Early Time Frame feature if it is greater than 0.5 or not;
   - Evidence: Look at the Rating Deviation feature if it is equal to 1;

(4) Query: Has the review a normal content or suspicious one?
   - Evidence: Look at the review text if it contains '!';
   - Evidence: Look if the reviewer uses second personal pronouns or not;

(5) Query: Did the reviewer describe in the review text his feeling or opinion about a real experience with a product/service?
   - Evidence: Look at the reviewer's feature-based opinion if it is deviated from the majority of reviewing feature-based opinions.

4.2 Analysis and Design

Analysis and Design is the second broad step of the UMP-SW methodology. Once goals and evidences to achieve them are identified, modelling the entities, attributes, relationships, and applicable rules can be started. This step also specifies the semantics of the model. We rely on the UML diagram to present the semantic model of RSPO. The UML diagram in Fig. 3 depicts the entities, attributes and relations and describes the objects, attributes, and relationships necessary to represent the RSPO. As depicted in Fig. 3, two main categories of spam feature are defined, Behavior Feature and Linguistic Feature. Behavior Feature has in turn three sub-categories such as *Profile Behavior Feature*, *User Behavior Feature* and *Review Behavior Feature*. The *Linguistic Feature* has also three sub-categories, *User Linguistic Feature*, *Review Linguistic Feature*, and *Opinion Deviation Feature*. Two possible linguistic values of spamicity level are defined: low and high. The *SpamicityLevel* class has four *has-Type* relations since each profile has a spamicity level, each user has a spamicity level and each review has two spamicity levels, the first one is based on review-features and the second presents the overall spamicity level which is based on the aggregation of the other spamicity levels. This provides a starting point to actually define entities/concepts of the probabilistic ontology. Since UML has a poor support to complex rule definitions required for uncertainty, the probabilistic rules are

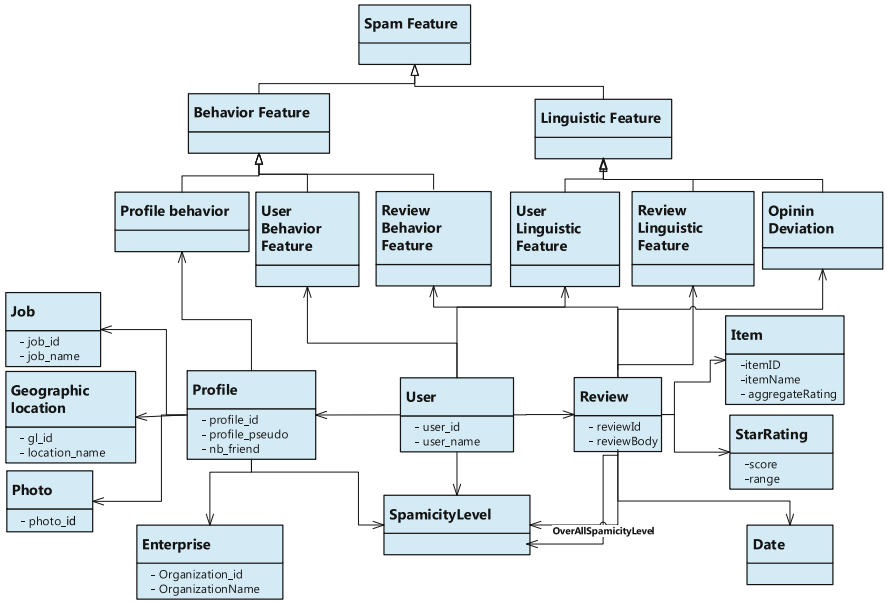

Fig. 3. UML diagram for review spam probabilistic ontology

specified separately. These rules are very useful when implementing the model in PR-OWL to specify the LPDs. Examples of the probabilistic rules required for the RSPO are presented as follows:

- If the majority of reviewer' star rates is between 1 and 2 then it is more likely that he tends to defame businesses which are competitor. Indeed, we can consider him as a spammer reviewer. At the same time, if he has an authentic profile then it is more likely to be a credible reviewer.
- If a review opinion agrees with the majority of reviews' opinions reviewing the same item, then it is more likely to be a credible review. Meanwhile, if its rating deviates from the average ratings then it is more likely to be a spam review.

Such probabilistic rules model the uncertain knowledge. These rules help in establishing causal relation between random variables.

4.3 Implementation

This phase starts by choosing the modelling language for the probabilistic ontology. In this work we use PR-OWL 2, which is supported by the UnBBayes PR-OWL 2 Plugin [20]. The entities, their attributes, and relations identified earlier are mapped to PR-OWL/MEBN constructs. The first step to go through is to map the entities, their attributes, and relations to PR-OWL, which uses essentially MEBN terms. Once the entities are defined, the uncertain characteristics

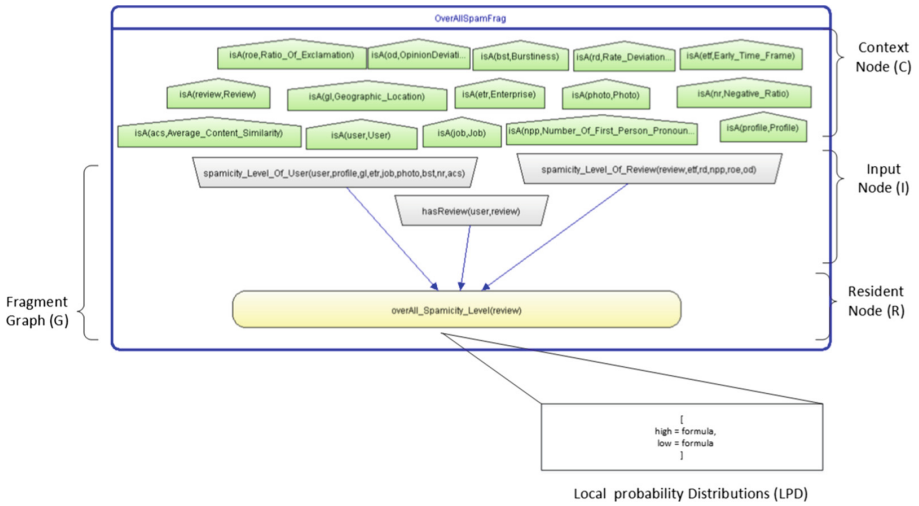

Fig. 4. MFrag for identifying the overall degree of review spamicity

should be identified. Uncertainty is represented in MEBN as random variables (RVs). In UnBBayes, an RV is first defined in its Home MFrag. Grouping RVs into MFrags closely follows the grouping observed in the Analysis and Design stage. Typically, an RV represents an attribute or a relation in the designed model. For instance, the RV *Spamicity_Level_Of_User* maps to the attribute *spamicityLevel* of the class *User* and the RV *hasProfile* maps to the relation hasProfile(Profile,User) (see Fig. 3). As a predicate relation, *hasProfile* relates a User to one Profile, the same way the class *Profile* is related to one User. Hence, the possible values (or states) of this RV are True or False. Each RV is represented as a resident node in its home MFrag. Once all resident RVs are created, their relations are defined by analyzing dependencies. This is achieved by looking at the rules defined in the semantic model of the RSPO. Rules consist in defining probability distribution of the resident node over its random variable instances. In our work, we define 21 MFrags including 21 resident nodes associated with their probability distributions. Figure 4 presents an MFrag example dealing with the overall spamicity level of a review. The resident node *overall_Spamicity_Level*(*review*) depends on three input nodes: the spamicity level of the reviewer, the spamicity level of the review and the relationship between the review and reviewer. The ordinary variables such as, user, review, profile, etc. can be filled in with different entities of type User, Review and Profile, etc. to make different instances of this MFrag as needed and reason about a specific situation. The local probability distributions of the defined MFrags are depicted in the next section.

4.4 Probability Distribution Definition

Once a random variable, its arguments, possible values, and respective mappings have been defined, it is necessary to define its probability distributions. We should define an LPD for each resident node (these local distributions apply only if all context nodes in the MFrag are satisfied). The main aim of the LPDs definition is to infer the overall spamicity level (OSL) of a given review. For doing so, Spamicity levels' LPDs for the different spam feature categories, namely SLP, SLUL, SLUB, SLU, SLRB, SLRL and SLR denoting Spamicity Level of Profile, Spamicity Level of User Linguistic, Spamicity Level of User Behavior, Spamicity Level of User, Spamicity Level of Review Behavior, Spamicity Level of Review Linguisitc and Spamicity Level of Review respectively, are defined according to the spam feature values (see Table 1). After that, we rely on the MEBN learning method [25] to learn the relationships between the spamicity level of spam feature categories and the spamicity level of the review in order to generate automatically the overall spamicity level LPD. The MEBN learning uses a relational model (RM) as a data schema for the dataset. The annotation of the review dataset is conducted in conjunction with cloud instructors from the IT department of the University of Sfax (considered as experts). The goal is to annotate the collected cloud service reviews from different SMPs (more details about the collected reviews are depicted in Sect. 5) with spam or credible reviews by relying on cloud service benchmarking tools, such as cloudHarmony [1] and Cloudlook [2]. To this end, the instructors organized themselves into four groups, where each group examined around 1000 reviews. Afterwards, they conducted a cross-validation process among the different groups.

4.5 RSPO Knowledge Base Population

The population of the RSPO is mainly based on three steps:

1. Data collection and pre-processing: this step consists in collecting and pre-processing information about user and review from SMPs (for more details the reader can refer to [8]).
2. Spam features detection: this step aims to detect and compute spam features of both users and collected reviews.
3. RSPO instantiation: this step instantiates automatically classes and relations using KARMA[1] tool, which is an information integration tool that enables users to quickly and easily integrate data from a variety of data sources. It maps structured sources to RSPO in order to build semantic descriptions.

4.6 Review Spam Probabilistic Ontology Reasoning

Once the probabilistic ontology is implemented and populated, it is possible to realize plausible reasoning through the process of creating a Situation-Specific

[1] http://usc-isi-i2.github.io/karma/.

Table 1. An excerpt of LPDs' definition.

MFrag name	LPD
SLUL	$if\ any\ user\ have\ (hasAverage\_Content\_Similarity = 1)\ [high = 0.9,\ low = 0.1]\ else\ [high = 0.1,\ low = 0.9]$
SLUB	$if\ any\ user\ have\ (hasBustiness = 1\ \&\ hasNegativeRatio = 1)\ [high = 0.9,\ low = 0.1]\ else\ [if\ any\ user\ have\ ((hasBustiness = 1\ \&\ hasNegativeRatio = 0)\ \mid\ (hasBustiness = 0\ \&\ hasNegativeRatio = 1))\ [high = 0.5,\ low = 0.5]\ else\ [high = 0.1,\ low = 0.9]]$
SLU	$if\ any\ user\ have\ (SLUP = high\ \&\ SLUB = high\ \&\ SLUL = high)\ [high = 0.9,\ low = 0.1]\ else\ [if\ any\ user\ have((SLUP = low\ \&\ SLUB = high\ \&\ SLUL = high)\ \mid\ (SLUP = high\ \&\ SLUB = low\ \&\ SLUL = high)\ \mid\ (SLUP = high\ \&\ SLUB = high\ \&\ SLUL = low))\ [high = 0.7,\ low = 0.3]\ else\ [if\ any\ user\ have((SLUP = low\ \&\ SLUB = low\ \&\ SLUL = high)\ \mid\ (SLUP = high\ \&\ SLUB = low\ \&\ SLUL = low)\ \mid\ (SLUP = low\ \&\ SLUB = high\ \&\ SLUL = low))\ [high = 0.3,\ low = 0.7]\ else\ [high = 0.1,\ low = 0.9]]]$

Bayesian Network (SSBN). UnBBayes has implemented an algorithm that creates an SSBN for a particular query. An example of reasoning is shown in Fig. 5. Information about review and reviewer are extracted from the provider DigitalOcean official Facebook page. As depicted in Table 2, the user's history information is missing. Consequently, we cannot compute neither the User Linguistic Features nor the User Behavior Features. In this case our approach does not consider these two categories in the spam judgment by given the same probability of the two values of the degree of being spam (high and low). Figure 5 presents the generated SSBN of the review. Given spam feature values, the inference system generates the probabilities of the two values of the degree of spamicity of the review: the probabilities of high and low. When returning to the example, the overall spamicity of the review is considered as high by 65.6% and as low by 34.4%.

Table 2. Examples of reviews. A: Available; NA: Not Available

Reviewer	Service	Rating	Socia media platform category	Content	Review information	User information	Profile information
mimi	DigitalOcean	5	SNS	Good!!	A	NA	A

5 Experiments and Results

This section presents the experimental evaluation part of this study including the datasets, the defined metrics as well as the obtained results.

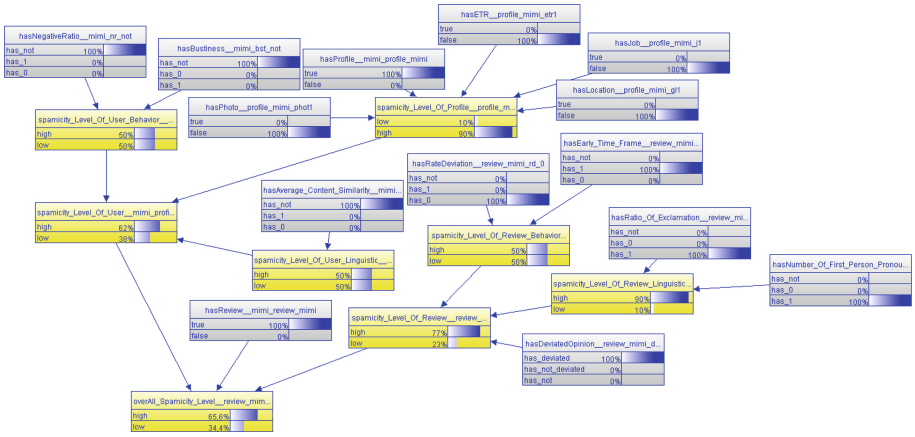

Fig. 5. SSBN for the query overall spamicity level of mimi's review

Table 3. Review datasets

	SNS	RLI	RF	All
#Reviews	1000	1000	1000	3000
% Spam reviews	20%	5%	13%	13%

5.1 Datasets and Evaluation Metrics

Datasets: Table 3 includes a summary of the used datasets and their characteristics. These datasets include the reviewers' impressions and comments about the quality of cloud services. As per this table, the datasets are categorized into three categories according to the review source:

- SNS dataset, includes reviews collected from Facebook pages as a SNS.
- Review platforms with LinkedIn authenticate dataset (RLI), includes reviews collected from online review platforms that obligate the access with LinkedIn account such as TrsutRadius [6] and G2Crowd [4].
- Review platforms Free dataset (RF), includes reviews collected from online review platforms when anyone can put a review without any authentication, such as hostadvice [5] and cloudReview [3].

We take 80% of the annotated reviews from each dataset category as a training dataset in order to learn the MEBN model and 20% are taken as test dataset to evaluate the effectiveness and the performance of the approach.

Evaluation Metrics: To evaluate the performance of our approach, four metrics are used: Precision (P), Recall (R), F1-score (F1) and Accuracy (A).

$$P = \frac{TP}{TP + FP} \tag{10}$$

$$R = \frac{TP}{TP + FN} \tag{11}$$

$$F1 = \frac{2 \times p \times R}{P + R} \tag{12}$$

$$A = \frac{TP + TN}{TP + FP + FN + TN} \tag{13}$$

In the context of this paper, the false positive (FP) refers to the number of credible reviews that are misidentified as spam ones, while the true positive (TP) refers to the number of correctly identified spam reviews. Similarly, the false negative (FN) refers to the number of spam reviews that are misidentified as credible ones, while the true negative (TN) refers to the number of correctly identified credible reviews.

5.2 Experimental Results

This section demonstrates the RSPO based approach effectiveness and performance.

(1) Overall effectiveness and performance analysis: Table 3 demonstrates that the RSPO based approach detects spam reviews from the three datasets (SNS, RLI and RF). In addition, Fig. 6 illustrates the RSPO performance in terms of precision, recall, F1-score and accuracy. As for this figure, the RSPO based approach has a high performance over the three datasets (around 90% for all metrics).

(2) Dataset impression on spam detection: Our experiments revealed a number of spam reviews in RLI dataset, but this number is much more important in SNS and RF. In fact, the RLI platforms, such as trustRadius, mainly verify the credibility of users (they mention "verified user"), who are obliged to use their LinkedIn identities prior to posting their reviews. Contrariwise, in Social Network, such as Facebook any person can create a fake profile and write a review. Table 3 shows the huge difference between the number of detected spam reviews and spammers found in review platforms and those of Facebook pages.

(3) Spam feature Analysis: The combination of Spam features can be a good hint for achieving better performance. The PA achieves better performance with RF dataset (around 90% of accuracy). Moreover, even in SNS dataset, the PA realizes a good result. In fact during our experiments, we noticed that when a spammer wants to promote a service (by giving a 5 star rate), he uses an authentic profile so that the profile authenticity feature cannot reflect the real state of the review (we found 24% of accuracy in this case). However, when he wants to defame a service, he hides his identity. Therefore, the PA in this case represents an important feature for the detection of spam (83% of accuracy). Besides, the opinion deviation feature (OD) achieves a greater influence on the performance of the spam review detection result in most datasets (especially for SNS and RLI datasets).

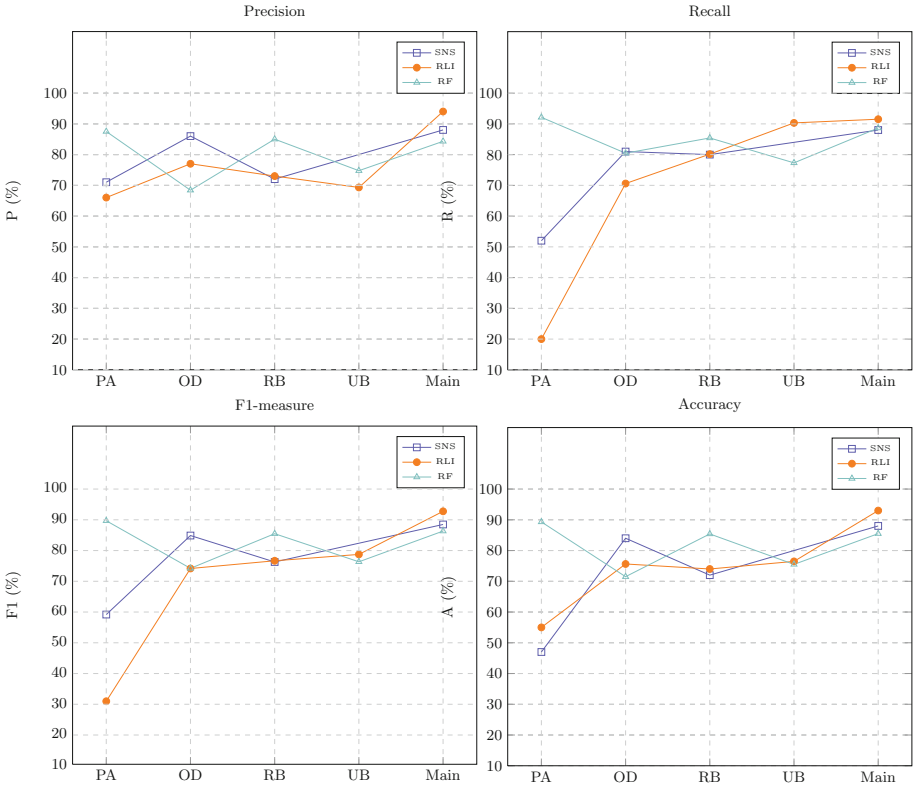

Fig. 6. Spam detection performance from different SMPs with different datasets. **RB:** Review Based features; **UB:** User Based features; **Main:** All spam features.

6 Related Work

In the last decade, a great number of research studies focus on the problem of spotting spammers and spam reviews. However, since the problem is non-trivial and challenging, it remains far from fully solved. To detect spammers, there are four categories of features in the literature, including review-behavioral, user-behavioral, review-linguistic, and user-linguistic.

Fei et al. in [15] consider the burstiness of each review to find spammers and spam reviews on Amazon. They build a network of reviewers appearing in different bursts. Then, they model reviewers and their co-occurrence in bursts as a Markov Random Field (MRF), and employ the Loopy Belief Propagation (LBP) method to infer whether a reviewer is a spammer or not. Shehnepoor et al. [27] propose a spam detection framework, namely NetSpam. This framework is established based on metapath concept and graph-based method to label reviews. The authors also introduce the importance of spam features to obtain better results on Yelp and Amazon Web sites. Xue et al. in [29] use the rate

Table 4. Analysis of the spam review detection approaches

Study	Review information	User history	SMP	Support of incomplete information
[15]		√	Amazon	
[27]	√	√	Yielp	
[26,29]		√	Yielp	
[16]	√	√	Amazon	
[24]	√	√	TripAdvisor, Yielp	√

deviation of a specific user and employs a trust-aware model to find the relationship between users to compute the final spamicity score. Savage et al. in [26], in turn, use the rate deviation to identify opinion spammers. They focus on the differences between user rating and the majority of honest users using a binomial model. Xie et al. in [28] use a temporal pattern (time window) to find singleton reviews (reviews written just once) on Amazon. Further, Heydari [16] proposes a spam detection system which investigates rate deviation, content based factors and activeness of reviewers in suspicious time intervals captured from time series of reviews by a pattern recognition technique. Mukherjee et al. [24] present a consistency model using limited information for detecting non-credible reviews. To do so, they rely on latent topic models leveraging review texts, item ratings, and timestamps. The above spam review detection approaches are summarized in Table 4. Compared to the existing works, the RSPO based approach covers the almost SMP including SNS and review platforms which was obviously not the case for the other approaches. This is by adding profile authenticity feature. Moreover, unlike the proposed works, our approach deals with missing information and uncertainty about spam judgment using a probabilistic reasoning.

7 Conclusion

Spam review is a continuing problem for consumers looking to be guided by online reviews in making their purchasing decisions. In the current study, we have introduced a Review Spam Probabilistic ontology (RSPO) based approach, which describes relevant concepts for the detection of spammer users and spam reviews from any social media platform. In addition, the RSPO can infer the degree of spam given incomplete information about spam features thanks to the probabilistic reasoning. In order to outperform the spam review detection, we extended the spam features with two new ones, *profile authenticity* and *opinion deviation* features. The experiments showed the improvement achieved by these two features. Moreover, they demonstrated the performance and the effectiveness of the RSPO based approach for the spam review detection from real data extracted from different categories of SMPs. As a future endeavor, we plan to

investigate the presented spam review detection approach by proposing a credible cloud service recommendation approach through online reviews.

References

1. Cloudharmony. cloudharmony.com
2. Cloudlook. www.cloudlook.com
3. Cloudreviews. cloudreviews.com
4. G2crowd. g2crowd.com
5. Hostadvice. hostadvice.com
6. Trustradius (2013). trustradius.com
7. Social media use in 2018 (2018). http://www.pewinternet.org/2018/03/01/social-media-use-in-2018/
8. Ben-Abdallah, E., Boukadi, K., Hammami, M.: SMI-based opinion analysis of cloud services from online reviews. In: Abraham, A., Muhuri, P.K., Muda, A.K., Gandhi, N. (eds.) ISDA 2017. AISC, vol. 736, pp. 683–692. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-76348-4_66
9. Carvalho, R.: Probabilistic ontology: representation and modeling methodology, January 2011
10. Carvalho, R., Laskey, K.B., Costa, P., Ladeira, M., Santos, L.L., Matsumoto, S.: Unbbayes: modeling uncertainty for plausible reasoning in the semantic web (2012)
11. Carvalho, R.N., Laskey, K.B., Costa, P.C.: PR-OWL: a language for defining probabilistic ontologies. Int. J. Approx. Reason. **91**, 56–79 (2017). https://doi.org/10.1016/j.ijar.2017.08.011, http://www.sciencedirect.com/science/article/pii/S0888613X17301044
12. Ding, Z., Peng, Y., Pan, R.: BayesOWL: uncertainty modeling in semantic web ontologies. In: Ma, Z. (ed.) Soft Computing in Ontologies and Semantic Web, pp. 3–29. Springer, Heidelberg (2006). https://doi.org/10.1007/978-3-540-33473-6_1
13. Esuli, A., Sebastiani, F.: Sentiwordnet: a publicly available lexical resource for opinion mining. In: Proceedings of the 5th Conference on Language Resources and Evaluation (LREC 2006), pp. 417–422 (2006)
14. Fei, G., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., Ghosh, R.: Exploiting burstiness in reviews for review spammer detection, pp. 175–184, January 2013
15. Fei, G., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., Ghosh, R.: Exploiting burstiness in reviews for review spammer detection. In: ICWSM (2013)
16. Heydari, A., Tavakoli, M., Salim, N.: Detection of fake opinions using time series. Expert Syst. Appl. **58**(C), 83–92 (2016). https://doi.org/10.1016/j.eswa.2016.03.020, https://doi.org/10.1016/j.eswa.2016.03.020
17. Jiang, S.Y., Yang, A.M.: Framework of clustering-based outlier detection. In: 2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery, vol. 1, pp. 475–479, August 2009. https://doi.org/10.1109/FSKD.2009.94
18. Laskey, K.B., Mahoney, S.M.: Network engineering for agile belief network models. IEEE Trans. Knowl. Data Eng. **12**(4), 487–498 (2000). https://doi.org/10.1109/69.868902
19. Li, F., Huang, M., Yang, Y., Zhu, X.: Learning to identify review spam. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, IJCAI 2011, vol. 3, pp. 2488–2493, AAAI Press (2011). https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-414

20. Matsumoto, S., et al.: UnBBayes: a Java framework for probabilistic models in AI (2011)
21. Mukherjee, A., Venkataraman, V., Liu, B., Glance, N.: What yelp fake review filter might be doing?, pp. 409–418, January 2013
22. Mukherjee, A., et al.: Spotting opinion spammers using behavioral footprints. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, pp. 632–640. ACM, New York (2013). https://doi.org/10.1145/2487575.2487580
23. Mukherjee, A., Venkataraman, V., Liu, B., Glance, N.S.: What yelp fake review filter might be doing? In: ICWSM (2013)
24. Mukherjee, S., Dutta, S., Weikum, G.: Credible review detection with limited information using consistency features. In: Frasconi, P., Landwehr, N., Manco, G., Vreeken, J. (eds.) ECML PKDD 2016. LNCS (LNAI), vol. 9852, pp. 195–213. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46227-1_13
25. Park, C.Y., Laskey, K.B., Costa, P.C.G., Matsumoto, S.: Multi-entity Bayesian networks learning for hybrid variables in situation awareness. In: Proceedings of the 16th International Conference on Information Fusion, pp. 1894–1901, July 2013
26. Savage, D., Zhang, X., Yu, X., Chou, P., Wang, Q.: Detection of opinion spam based on anomalous rating deviation. Expert Syst. Appl. **42**(22), 8650–8657 (2015). https://doi.org/10.1016/j.eswa.2015.07.019, http://www.sciencedirect.com/science/article/pii/S0957417415004790
27. Shehnepoor, S., Salehi, M., Farahbakhsh, R., Crespi, N.: NetSpam: a network-based spam detection framework for reviews in online social media. IEEE Trans. Inf. Forensics Secur. **12**(7), 1585–1595 (2017). https://doi.org/10.1109/TIFS.2017.2675361
28. Xie, S., Wang, G., Lin, S., Yu, P.S.: Review spam detection via temporal pattern discovery. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2012, pp. 823–831. ACM, New York (2012). https://doi.org/10.1145/2339530.2339662
29. Xue, H., Li, F., Seo, H., Pluretti, R.: Trust-aware review spam detection. In: 2015 IEEE Trustcom/BigDataSE/ISPA, vol. 1, pp. 726–733, August 2015. https://doi.org/10.1109/Trustcom.2015.440
30. Yang, Y., Calmet, J.: OntoBayes: an ontology-driven uncertainty model. In: International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC 2006), vol. 1, pp. 457–463, November 2005. https://doi.org/10.1109/CIMCA.2005.1631307