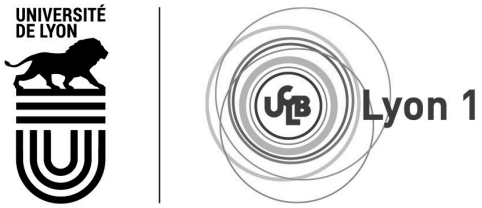
Approche robuste pour l'évaluation de la confiance des ressources sur le Web

Zohra Saoud



N° d'ordre NNT : 2016LYSE1331

THÈSE DE DOCTORAT DE L'UNIVERSITÉ DE LYON

opérée au sein de

l'Université Claude Bernard Lyon 1

École Doctorale ED 512

InfoMaths

Spécialité de doctorat : Informatique

Soutenue publiquement à le 14/12/2016, par :

Zohra Saoud

Approche robuste pour l'évaluation de la confiance des ressources sur le Web

---

Devant le jury composé de :

Marinette SAVONNET, Maître de conférences - HDR, Université de Bourgogne	Rapporteur(e)
Ahmed LBATH, Professeur des Universités - HDR, Université de GRENOBLE	Rapporteur
Daniela GRIGORI, Professeur des Universités - HDR, Université Paris Dauphine	Examinatrice
Abdelkader HAMEURLAIN, Professeur des Universités - HDR, Université Paul Sabatier à Toulouse	Président
Ladjel BELLATRECHE, Professeur des Universités - HDR, Ecole ENSMA à Poitiers	Examinateur
Djamal Benslimane, Professeur des Universités - HDR, Université Claude Bernard Lyon 1	Directeur de thèse
Noura FACI, Maitre de conférences, Université Claude Bernard Lyon 1	Co-directrice de thèse

Sommaire

Liste des figures

Liste des tableaux

Abstract

This thesis in Computer Science is part of the trust management field and more specifically recommendation systems. These systems are usually based on users' experiences (i.e., qualitative / quantitative) interacting with Web resources (eg. Movies, videos and Web services). Recommender systems are undermined by three types of uncertainty that raises due to users' ratings and identities that can be questioned and also due to variations in Web resources performance at run-time. We propose a robust approach for trust assessment under these uncertainties.

The first type of uncertainty refers to users' ratings. This uncertainty stems from the vulnerability of the system in the presence of malicious users providing false ratings. To tackle this uncertainty, we propose a fuzzy model for users' credibility. This model uses a fuzzy clustering technique to distinguish between malicious users and strict users usually excluded in existing approaches.

The second type of uncertainty refers to user's identity. Indeed, a malicious user purposely creates virtual identities to provide false ratings. To tackle this type of attack known as Sybil, we propose a ratings filtering model based on the users' credibility and the trust graph to which they belong. We propose two mechanisms, one for assigning capacities to users and the second one is for selecting users whose ratings will be retained when evaluating trust. The first mechanism reduces the attack capacity of Sybil users. The second

mechanism chose paths in the trust graph including trusted users with maximum capacities. Both mechanisms use users' credibility as heuristic.

To deal with the uncertainty over the capacity of a Web resource in satisfying users' requests, we propose two approaches for Web resources trust assessment, one deterministic and one probabilistic. The first consolidates users' ratings taking into account users credibility values. The second relies on probability theory coupled with possible worlds semantics. Probabilistic databases offer a better representation of the uncertainty underlying users' credibility and also permit an uncertain assessment of resources trust.

Finally, we develop the system WRTrust (Web Resource Trust) implementing our trust assessment approach. We carried out several experiments to evaluate the performance and robustness of our system. The results show that trust quality has been significantly improved, as well as the system's robustness in presence of false ratings attacks and Sybil attacks.

Keywords : Web resource, trust, credibility, fuzzy clustering, probability.

Résumé

Cette thèse en Informatique s'inscrit dans le cadre de gestion de la confiance et plus précisément des systèmes de recommandation. Ces systèmes sont généralement basés sur les retours d'expériences des utilisateurs (i.e., qualitatifs/quantitatifs) lors de l'utilisation des ressources sur le Web (ex. films, vidéos et service Web). Les systèmes de recommandation doivent faire face à trois types d'incertitude liés aux évaluations des utilisateurs, à leur identité et à la variation des performances des ressources au fil du temps. Nous proposons une approche robuste pour évaluer la confiance en tenant compte de ces incertitudes.

Le premier type d'incertitude réfère aux évaluations. Cette incertitude provient de la vulnérabilité du système en présence d'utilisateurs malveillants fournissant des évaluations biaisées. Pour pallier cette incertitude, nous proposons un modèle flou de la crédibilité des évaluateurs. Ce modèle, basé sur la technique de clustering flou, permet de distinguer les utilisateurs malveillants des utilisateurs stricts habituellement exclus dans les approches existantes.

Le deuxième type d'incertitude réfère à l'identité de l'utilisateur. En effet, un utilisateur malveillant a la possibilité de créer des identités virtuelles pour fournir plusieurs fausses évaluations. Pour contrecarrer ce type d'attaque dit Sybil, nous proposons un modèle de filtrage des évaluations, basé sur la crédibilité des utilisateurs et le graphe de confiance auquel ils appartiennent. Nous proposons

deux mécanismes, l'un pour distribuer des capacités aux utilisateurs et l'autre pour sélectionner les utilisateurs à retenir lors de l'évaluation de la confiance. Le premier mécanisme permet de réduire le risque de faire intervenir des utilisateurs multi-identités. Le second mécanisme choisit des chemins dans le graphe de confiance contenant des utilisateurs avec des capacités maximales. Ces deux mécanismes utilisent la crédibilité des utilisateurs comme heuristique.

Afin de lever l'incertitude sur l'aptitude d'une ressource à satisfaire les demandes des utilisateurs, nous proposons deux approches d'évaluation de la confiance d'une ressource sur le Web, une déterministe et une probabiliste. La première consolide les différentes évaluations collectées en prenant en compte la crédibilité des évaluateurs. La deuxième s'appuie sur la théorie des bases de données probabilistes et la sémantique des mondes possibles. Les bases de données probabilistes offrent alors une meilleure représentation de l'incertitude sous-jacente à la crédibilité des utilisateurs et permettent aussi à travers des requêtes un calcul incertain de la confiance d'une ressource.

Finalement, nous développons le système WRTrust (Web Resource Trust) implémentant notre approche d'évaluation de la confiance. Nous avons réalisé plusieurs expérimentations afin d'évaluer la performance et la robustesse de notre système. Les expérimentations ont montré une amélioration de la qualité de la confiance et de la robustesse du système aux attaques des utilisateurs malveillants.
Mots clés : ressource sur le Web, confiance, crédibilité, clustering flou, attaque Sybil.

Chapitre 1

Introduction

Sommaire

1.1 Cadre de la thèse

Dans cette section, nous présentons le cadre de la thèse : son contexte général et ses motivations.

1.1.1 Contexte général

Le Web a complètement révolutionné notre vie quotidienne donnant accès à diverses ressources telles que le téléchargement de musique, la consultation des actualités en ligne, l'achat de billets de transport, etc. Plus généralement, une ressource se définit comme toute entité susceptible d'être identifiée, nommée, manipulée sur le Web ou dans tout système d'information utilisant les technologies du Web (ex., un document électronique, un service, une image ou en agrégeant d'autres ressources telles qu'un album photo est le résultat d'agrégation d'autres ressources, des images) [13]. Avec la croissance exponentielle du nombre de ressources disponibles sur le Web, l'utilisateur n'a jamais eu autant de choix. Cependant, des recherches dans le domaine du marketing ont démontré que l'abondance de choix rend la prise de décision plus difficile [36]. Réduire les choix mis à disposition des utilisateurs permettra alors d'accélérer leur prise de décision. Ainsi, le problème, n'est plus de trouver les ressources, mais plutôt de cibler l'information et éliminer celles qui peuvent être inutiles. Pour ce faire, il est nécessaire de trouver des informations concernant les ressources afin de pouvoir les comparer par la suite. Le Web n'a pas seulement changé nos habitudes commerciales mais également sociales telles que nos méthodes de travail, nos modes de communications, et même nos loisirs. En effet, avec l'apparition des applications du Web 2.0 et des réseaux sociaux (Facebook, LinkedIn, Twitter, etc.), l'internet est très vite passé d'un réseau d'ordinateurs interconnectés à un réseau d'utilisateurs actifs qui collaborent entre eux. Les utilisateurs ont alors la possibilité de partager leurs avis (ou opinions) sur différentes ressources avec des millions de personnes généra-

lement inconnues. L'utilisateur n'est plus uniquement un simple consommateur d'informations, mais également un acteur produisant à son tour le contenu à travers les commentaires, les discussions dans les forums, les blogs, etc. Les utilisateurs du Web se trouvent alors confrontés à une abondance d'informations concernant les ressources. Lorsque ces derniers souhaitent comparer les prix d'un produit ou d'un billet d'avion ou recherchent des critiques sur des films, les informations disponibles sur les ressources excédent ce dont ils ont besoin pour faire leur choix. Face à l'afflux des informations disponibles, l'utilisateur a l'embarras du choix. L'indécision de l'utilisateur est d'autant amplifiée par le risque élevé de fraudes ou d'inexactitude des avis partagés. Ce risque est principalement lié à l'ouverture de comptes anonymes dans les communautés de partage d'avis comme dans les sites marchands eBay et Amazon.

Afin de renforcer la confiance de l'utilisateur dans les ressources sur le Web et le guider dans ses choix, deux types d'approches ont été proposées pour la découverte des ressources sur le Web. Le premier type d'approche se base sur la description non fonctionnelle de la ressource. Par exemple pour le cas des services Web, cette description correspond à la qualité de services (QoS) publiée dans des registres centralisés par les fournisseurs de ces services (ex., [23]). Le deuxième type d'approche concerne la collecte des retours d'expériences des utilisateurs (i.e., qualitatifs/quantitatifs) sur l'utilisation des ressources en question. Ces retours sont des évaluations reflétant la satisfaction des utilisateurs relative au service fourni par une ressource. Néanmoins, se fier aux fournisseurs et/ou utilisateurs soulève chez les futurs utilisateurs de ces ressources le questionnement suivant : *quelle confiance attribuer aux ressources en présence de sur/sous-enchères intentionnelles de la qualité de service des ressources ?* En effet, un utilisateur peut vouloir promouvoir une ressource et rétrograder une autre.

Le critère de la confiance représente alors un facteur clé dans la

sélection des ressources sur le Web. La confiance se définit comme étant la fiabilité d'une ressource à fournir les fonctionnalités et la qualité de service annoncées. Dans la littérature, nous distinguons deux types de modèles de confiance pour évaluer le critère de confiance. Le premier type de modèle utilise des évaluations fournies par des utilisateurs pour calculer la confiance des ressources sur le Web (ex., [48]). Quant au second type de modèle, il repose sur l'observation du comportement des ressources au cours d'une période donnée (ex., [44]). Cependant, ce caractère dynamique ne caractérise pas toutes les ressources sur le Web telles que les films ou les articles. De plus, avec la nouvelle tendance des réseaux sociaux, ce type de modèle est moins susceptible de convaincre les utilisateurs actuels qui prêtent plus d'importance aux avis des autres utilisateurs. Par conséquent, nous nous intéressons, dans cette thèse, plus particulièrement au premier type de modèle de confiance permettant de résoudre le problème d'abondance des informations sur les ressources sur le Web.

1.1.2 Motivations

Plusieurs systèmes d'évaluation de la confiance ont été présentés dans la littérature. Durant plusieurs années, le principal aspect sur lequel se sont focalisés les chercheurs fût l'amélioration du critère de performance relatif à la qualité de la valeur de confiance calculée. Cependant, les résultats obtenus par ces systèmes ne sont pas toujours corrects en présence de tentatives malveillantes pour influencer l'évaluation de la confiance. Ces tentatives sont faites par les utilisateurs isolés ou en groupe afin de détourner le système et d'induire ses utilisateurs en erreur. Nous citons quelques exemples d'attaques comme les fausses évaluations et l'évaluation multiple d'un même produit en utilisant des fausses identités, etc. La vulnérabilité des systèmes de confiance et de réputation à ce genre d'attaques limite leur efficacité et leur utilisation. La robustesse du système s'avère alors un critère encore plus important que la

performance [40]. Plusieurs problèmes relatifs à la robustesse ont été soulevés et sont considérés actuellement comme des challenges toujours ouverts et doivent être résolus.

Le premier type d'attaques auquel nous nous intéressons est l'attaque des évaluations biaisées/fausses. Ces dernières représentent des avis augmentant ou diminuant de manière intentionnelle et à des fins malveillantes les valeurs des propriétés non-fonctionnelles des ressources sur le Web. Pour traiter les évaluations biaisées, des approches telles que Cloud Armor [29] et RateWeb [26], considèrent la crédibilité de l'utilisateur lors du calcul de la confiance. D'autres approches associent la crédibilité de l'utilisateur à son niveau d'expertise [33]. Lorsque les utilisateurs sont en désaccord vis-à-vis de l'évaluation d'une ressource, ces approches établissent un consensus en utilisant l'avis de la majorité ou la moyenne des évaluations. Les utilisateurs proches de l'avis de la majorité ou la moyenne sont plus crédibles que ceux s'en trouvant éloignés. La variation du niveau de fiabilité ou d'expertise des utilisateurs pose également un problème au moment de l'évaluation de la confiance. Prendre en compte les évaluations collectées de tous les utilisateurs sans considérer ces deux critères peut fausser le calcul de la confiance des ressources.

Le deuxième type d'attaques est l'attaque des utilisateurs "Sybil". Elles réfèrent à des utilisateurs qui créent délibérément des identités virtuelles et par conséquent, fournissent de fausses évaluations concernant les ressources sur le Web. Il s'agit d'un problème récurrent dans les environnements en ligne où l'inscription au système est à coût très faible ou nul, comme c'est le cas des sites commerciaux tel que eBay et Amazon. Afin de palier à ce problème, des approches de confiance basées sur les réseaux sociaux (ex., [51] et [52]) procèdent à une étape de filtrage des évaluations avant d'évaluer la confiance. En effet, les utilisateurs Sybil, vu qu'ils se dissimulent derrière des fausses identités, ils ont peu de chance de gagner la confiance des utilisateurs réels (non Sybil). Ils arrivent

donc à créer un nombre limité de liens avec eux. L'analyse de la structure/typologie du réseau social permettra de repérer ces utilisateurs.

1.2 Problématique

La problématique principale de nos travaux de recherches est formulée de la manière suivante :

Comment évaluer la confiance des ressources sur le Web d'une manière robuste et efficace en dépit des attaques des utilisateurs malveillants ?

L'évaluation de la confiance des ressources sur le Web se base sur les évaluations fournies par les utilisateurs ayant déjà une expérience d'utilisation de la ressource et disposant d'un avis sur cette dernière. Notre but consiste alors à aider les futurs utilisateurs à prendre une décision avant de s'engager dans une interaction avec une ressource. Or, les utilisateurs ayant peu ou pas d'expériences ne sont pas aptes à fournir des valeurs appropriées de la confiance des ressources sur le Web. Il s'agit alors d'aider ces utilisateurs à prendre une décision sûre ou tout simplement raisonnable alors que nous ne disposons que d'informations incertaines sur la ressource. Ainsi, nous classifions les questions de recherche abordées dans cette thèse selon le type d'incertitude auquel on peut faire face lors de l'évaluation des ressources sur le Web :

- Le premier type d'incertitude réfère aux évaluations. (U_1). U_1 est le résultat de l'incohérence des évaluations fournies par les utilisateurs au fil du temps. Cette incertitude provient de la vulnérabilité du système à l'attaque des évaluations biaisées. Une solution pour pallier ce type d'incertitude est d'inclure la crédibilité des utilisateurs lors du calcul de la confiance. Cependant, les approches

existantes de confiance basées sur la crédibilité supposent que les utilisateurs ont soit une bonne expertise, soit une certaine fiabilité. Alors que la crédibilité elle-même doit être évaluée en prenant compte l'expertise et la fiabilité qui sont considérés, par définition, comme ses principales composantes. Nous étudions alors :

Q_1. Comment évaluer efficacement la crédibilité des utilisateurs en prenant en compte leur fiabilité et leur expertise ?

- Le deuxième type d'incertitude réfère à l'identité de l'utilisateur (U_2). U_2 résulte de la possibilité de créer des identités virtuelles dans le système utilisées afin d'émettre de fausses évaluations. Cela correspond à l'attaque Sybil. Les approches proposées pour contrecarrer ce type d'attaques filtrent les évaluations afin de repérer les utilisateurs Sybil. Toutefois, elles se concentrent seulement sur l'identité des utilisateurs sans prendre en compte la qualité des évaluations fournies par ces derniers et entre autres, leurs crédibilités. Nous étudions alors :

Q_2. Comment collecter les évaluations afin d'avoir un nombre maximal d'utilisateurs crédibles et non Sybil ?

- Le troisième type d'incertitudes réfère à l'aptitude d'une ressource à satisfaire les demandes des utilisateurs (U_3). U_3 résulte essentiellement de l'incohérence des valeurs de la qualité de service induite par la nature dynamique de la ressource et/ou à un comportement malveillant venant du fournisseur de la ressource. Un utilisateur fait confiance à une ressource si cette dernière satisfait de manière significative un grand nombre de ses demandes. Une solution pour contrecarrer ce type d'incertitude est d'évaluer la confiance de la ressource. Nous nous demandons alors :

Q_3. Comment agréger les évaluations sachant que les utilisateurs

n'ont pas tous le même niveau de crédibilité et certains peuvent avoir des identités virtuelles ?

1.3 Contributions

Dans ce qui suit, nous présentons les principales contributions scientifiques de cette thèse. Notre objectif principal est de proposer une approche holistique d'évaluation de la confiance des ressources sur le Web, qui soit robuste contre les évaluations biaisées et les attaques Sybil. Les trois premières contributions sont des modèles que nous proposons en réponse aux trois questions de recherche Q_1, Q_2 et Q_3. La quatrième contribution correspond, quant à elle, à la mise en oeuvre et aux expérimentations de l'approche holistique. La figure 1.1 montre un aperçu global de notre approche d'évaluation de la confiance et les connexions entre les différents modèles proposés.

C_1. Modèle de crédibilité flou résilient aux évaluations biaisées

Pour répondre à Q_1 relative à la robustesse aux attaques des fausses évaluations (biaisées), nous proposons un nouveau modèle d'évaluation de la crédibilité des utilisateurs. Le modèle proposé se base uniquement sur les évaluations fournies par les utilisateurs, et n'exige aucune information sur leurs identités et leurs interactions, assurant une totale protection de leurs vies privées. La crédibilité est évaluée selon deux critères : la fiabilité et l'expertise. La fiabilité est calculée par rapport à la proximité de l'avis majoritaire. Néanmoins, un utilisateur expert n'aura aucun intérêt à s'aligner avec la majorité et il maintient son avis indépendamment de l'avis majoritaire, comme dans le cas où la majorité est composée de faux avis. Généralement, ces utilisateurs sont plus stricts dans leur évaluation de la ressource. Notre objectif principal est d'inclure les avis des utilisateurs présentant à la fois ces deux caractéristiques et qui sont

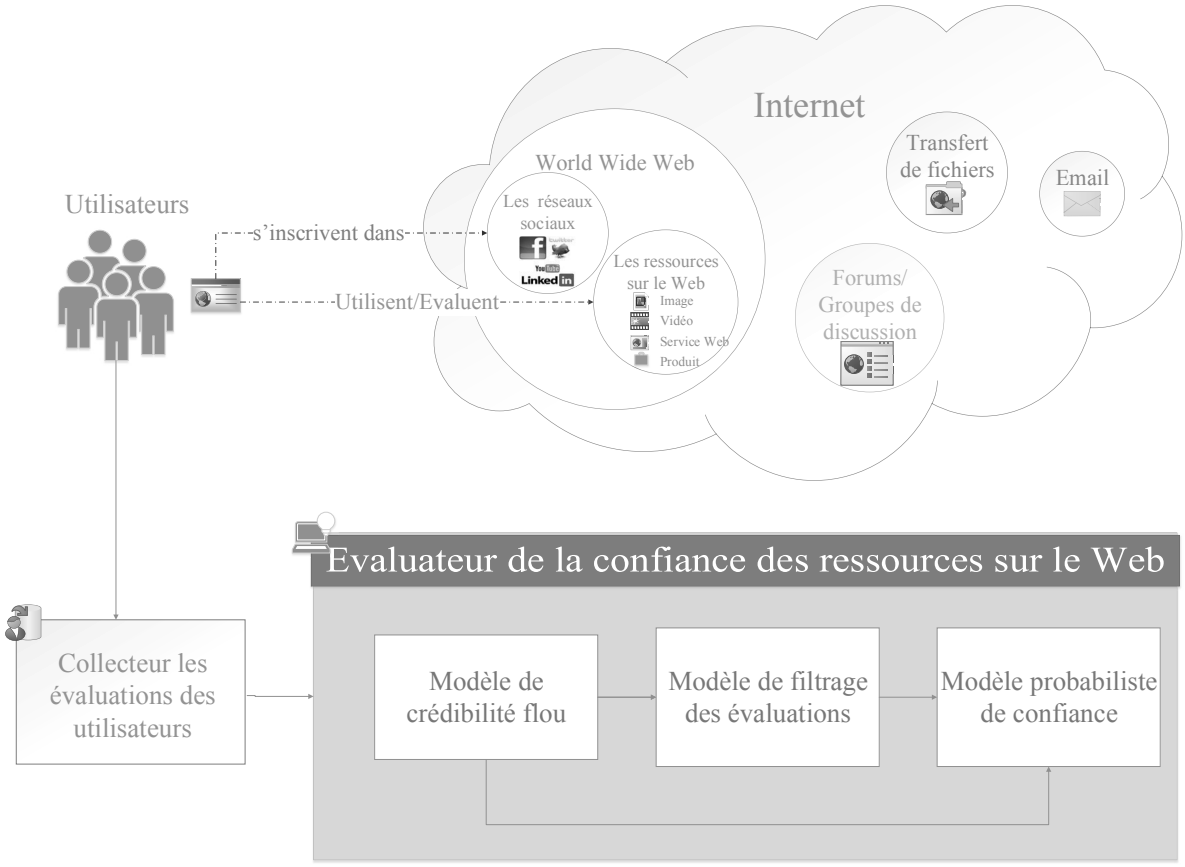


FIGURE 1.1 – Aperçu global de l'approche d'évaluation de la confiance

souvent négligées dans les approches classiques. Ceci permettra ainsi de réduire l'écart qui pourrait exister entre les évaluations de ces utilisateurs et l'avis actuel de la majorité. A cette fin, nous utilisons une technique de clustering flou des évaluations. Dans cette technique l'incertitude est représentée par des frontières graduelles entre les groupes d'avis à la place de frontières nettes entre eux. Elle s'exprime par un degré d'appartenance à une évaluation et à un ou plusieurs groupes. Ainsi, les évaluations situées à la frontière du groupe majoritaire et qui peuvent correspondre à des évaluations des utilisateurs stricts seront incluses dans ce dernier. Afin de déterminer le groupe majoritaire nous proposons trois stratégies en utilisant des valeurs qualitatives du degré d'appartenance à un cluster flou : stratégie faible, stratégie modérée, et stratégie forte.

C_2. Modèle de filtrage des évaluations tolérant aux attaques Sybil

Nous proposons un modèle de filtrage des évaluations afin de répondre à Q_2 concernant la robustesse de notre système contre les attaques Sybil. Le modèle combine l'évaluation de la crédibilité proposée dans C1 avec une analyse de la structure du réseau social. Le but principal est de sélectionner parmi les évaluations collectées celles fournies par des utilisateurs crédibles ayant une seule identité. Ces utilisateurs sont plus susceptibles de fournir des évaluations cohérentes et par conséquent, dignes de confiance. Comme il s'agit d'utilisateurs réels et honnêtes, ils n'établissent des liens avec d'autres utilisateurs du réseau social seulement s'il existe un lien de confiance véritable entre eux. Afin d'éviter les utilisateurs Sybil, le modèle collecte les évaluations en partant d'un utilisateur source connu et donc fiable et en explorant ses liens sociaux. Nous proposons un mécanisme de distribution des capacités. En effet, la capacité distribuée correspond au pouvoir d'évaluation que l'on souhaite accorder aux utilisateurs selon leurs crédibilités. En se ba-

sant sur des capacités distribuées, nous sélectionnons un ensemble d'utilisateurs crédibles et ayant une seule identité. Cette sélection est réalisée en utilisant l'algorithme de flot maximal.

C_3. Modèle déterministe et probabiliste de confiance

Nous proposons deux approches d'évaluation de la confiance d'une ressource sur le Web, une déterministe et une probabiliste pour répondre à Q_4 concernant l'agrégation des évaluations sachant que les utilisateurs n'ont pas tous le même niveau de crédibilité et certains peuvent avoir des identités virtuelles. La première approche vient consolider les évaluations des utilisateurs en prenant en compte la crédibilité des évaluateurs. La prise en compte de la crédibilité de l'utilisateur permet de résoudre le problème d'incohérence des évaluations. Dans notre approche, plus les utilisateurs sont crédibles plus leurs évaluations impacteront le calcul de la confiance. Cette incohérence se manifeste par une divergence des avis des utilisateurs. Troffaes a démontré que le recours à des probabilités permet de formaliser le problème de divergence d'avis [43]. Nous associons alors dans la deuxième approche la notion de crédibilité à celle de la probabilité. Les bases de données probabilistes associées à la sémantique de la théorie des mondes possibles constituent une solution intéressante au calcul de la confiance d'une ressource. Elles ont été largement utilisées pour représenter et analyser les données incertaines (extraction de données, etc.). Dans une base de données probabiliste chaque donnée (ou tuple) a une certaine probabilité d'appartenir à la base de données. Les bases de données probabilistes offrent alors une meilleure représentation de l'incertitude sous-jacente à la crédibilité des utilisateurs et permettent aussi à travers des requêtes un calcul incertain de la confiance d'une ressource [7].

C_4. Mise en oeuvre et expérimentations de l'approche holis-

tique

Nous proposons une implémentation de notre approche d'évaluation de la confiance des ressources sur le Web à travers le système WRTrust (Web Resource Trust). Nous avons menés plusieurs expérimentations afin d'évaluer les différents modèles et concepts proposés en termes de performance et de robustesse du système WRTrust. Afin d'évaluer le premier critère, nous avons utilisé des métriques telles que l'erreur moyenne de la valeur de confiance (RMSE), la précision et le rappel. Et afin d'évaluer le deuxième critère, nous avons simulé des comportements malveillants tels que l'émission d'évaluations biaisées ou la création de fausses identités.

1.4 Structure du manuscrit

Ce manuscrit est composé de 6 chapitres en complément de cette introduction.
Dans le chapitre 2 nous définissons tout d'abord les principaux concepts relatifs au sujet de la thèse. Nous présentons par la suite les études qui analysent les systèmes de confiance afin de mieux se positionner par rapport aux travaux existants. Enfin nous présentons des travaux de recherche existants dans le domaine de l'évaluation de la confiance les plus proches de notre travail.
Dans le chapitre 3, nous détaillons un modèle de crédibilité flou résilient aux attaques des fausses évaluations. Le modèle proposé prend en compte à la fois la fiabilité et l'expertise de l'utilisateur lors de l'évaluation de la crédibilité. Nous discutons dans ce chapitre la motivation derrière l'utilisation du clustering flou sur laquelle se base notre modèle de crédibilité, et comment est estimée la crédibilité d'un utilisateur. Nous proposons enfin une approche déterministe d'évaluation de la confiance combinant l'ensemble des évaluations des utilisateurs en fonction de leurs crédibilités.

Dans le chapitre 4, nous proposons un modèle de filtrage des évaluations tolérant aux attaques Sybil. Nous définissons alors un mécanisme de distribution de capacités aux utilisateurs reflétant leurs capacités d'évaluation ainsi qu'un mécanisme de sélection des utilisateurs. Les deux techniques sont basées sur le modèle de crédibilité. Dans le chapitre 5, nous présentons un modèle probabiliste d'évaluation de la confiance des ressources sur le Web. Le modèle modélise l'ensemble des évaluations sous forme d'une base de données probabiliste et calcule la confiance comme une évaluation de requêtes probabilistes.
Dans le chapitre 6, nous présentons un système d'évaluation de la confiance fondé sur les différents modèles proposés. Nous avons exploité ce système pour mener plusieurs expérimentations analysent la robustesse et la performance de notre approche. Nous avons varié les expérimentations en utilisant plusieurs jeux de données d'évaluations de ressources sur le Web du monde réel (ex., MovieLens, WSDream, etc). Nous avons développé différentes interfaces graphiques pour répondre aux exigences de l'utilisateur (ex., le niveau de confiance).
Dans le chapitre 7, nous concluons notre travail en revenant sur les principales contributions apportées dans cette thèse ainsi que leurs limites et nous identifions enfin les perspectives de recherche.

Chapitre 2

État de l'art

Sommaire

2.1 Concepts fondamentaux

Dans cette section, nous définissons les concepts fondamentaux relatifs à la thèse afin de relever toute ambiguïté sur les termes utilisés dans ce manuscrit. Nous présentons ensuite les différents problèmes rencontrés par les systèmes de confiance et les différentes attaques réalisées contre ces systèmes tels qu'ils sont définis dans la littérature.

2.1.1 Définitions

L'objectif principal de la thèse est de proposer une solution pour l'évaluation de la confiance des ressources sur le Web, il est alors primordial de donner la définition d'une ressource sur le Web.

Définition 1 : une *ressource sur le Web* représente toute entité pouvant avoir un identifiant même en dehors de la portée du Web [13]. Le RFC (Request For Comments) 2396 définit une ressource sur le Web comme suit : " *A resource can be anything that has identity. Familiar examples include an electronic document, an image, a service (e.g., "today's weather report for Los Angeles"), and a collection of other resources. Not all resources are network "retrievable"; e.g., human beings, corporations, and bound books in a library can also be considered resources.* ".

Depuis des décennies, la notion de confiance a fait l'objet de plusieurs études dans divers domaines tel que l'économie, la politique, la sociologie, etc. Cependant, il est toujours difficile de lui attribuer une définition précise et communément admise. Ceci s'explique par le fait que la confiance est une notion multidimensionnelle : cognitive, affective/émotionnelle et comportementale, et elle est souvent associée à des termes tels que la crédibilité, la fiabilité, l'honnêteté, la sécurité et la confidence.

Deux variantes de la *confiance* sont identifiées dans la littérature : la confiance d'évaluation et la confiance de décision.

Définition 2 : la *confiance d'évaluation* est définie par Gambetta [10] comme "*... a particular level of the subjective probability with which an agent assesses that another agent or group of agents will perform a particular action, both before he can monitor such action (or independently of his capacity ever to be able to monitor it) and in a context in which it affects his own action*".

Cette variante de la confiance est celle adoptée par les systèmes d'évaluation de la confiance. En effet, il s'agit d'une vision calculatrice de la confiance qui correspond à une évaluation subjective d'un comportement anticipé, celui de la ressource dans notre approche.

Définition 3 : la *confiance de décision* est définie par McKnight [27] comme "*... the extent to which one intends to depend on a non-specific other party in a given situation*".

Elle représente la volonté d'un pair *A* de dépendre d'un pair *B* dans un contexte donné où il a le sentiment d'être en sécurité, même dans le cas d'éventuelles conséquences négatives. Elle sous-entend alors une décision d'accepter les risques liés à l'engagement avec l'autre pair. La confiance est justifiée lorsque le résultat/bénéfice attendu est atteint. Dans notre approche, *A* représente l'utilisateur et *B* peut représenter la ressource ou plus généralement le système de confiance. Cette variance de la confiance concerne plutôt la décision de l'utilisateur à interagir avec la ressource ou pas ou à prendre en compte l'évaluation de la ressource par le système.

Cette relation de dépendance est aussi présente entre le système de confiance et l'utilisateur (i.e, ses évaluations sont-elles plausibles/vraisemblables ?). Dans ce contexte, le terme utilisé est la crédibilité. Cette dernière se rapporte aux utilisateurs de la ressource, tandis que la confiance s'applique aux ressources.

Définition 4 : la *crédibilité* est définie par Bordens et Horowitz par deux composantes [2] : l'*expertise* qui provient des connaissances de l'utilisateur, son parcours, sa notoriété, etc. ; et la *fiabilité* qui réfère au "*... the audience's assessment of the communicator's character as well as his or her motives for delivering the message*".

L'évaluation de la confiance de la ressource et la crédibilité de l'utilisateur s'inscrivent dans un cadre plus général qui est la gestion de la confiance.

Définition 5 : la *gestion de la confiance*, elle est définie par Grandison comme "*... the activity of collecting, codifying, analyzing and evaluating evidence relating to competence, honesty, security or dependability with the purpose of making assessments and decisions regarding trust relationships*" [11].

Un système d'évaluation de la confiance peut être performant, c'est-à-dire remplissant les fonctionnalités demandées par l'utilisateur, mais ne pas fournir des résultats stables au cours du temps. Ceci est du à la présence d'utilisateurs malveillants essayant de détourner les résultats du système et d'induire ses utilisateurs en erreur. La stabilité de la performance du système se définit par la robustesse.

Définition 6 : La *robustesse* est définie, dans le glossaire standard de IEEE pour les terminologies de l'ingénierie logicielle, par "*The degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions*" [16].

La robustesse est un facteur de qualité important surtout lorsque les utilisateurs s'appuient essentiellement sur le système d'évaluation

de la confiance pour exécuter des applications critiques. Elle mesure principalement la capacité du système à atteindre ses objectifs malgré la présence d'éléments externes perturbateurs.

2.1.2 Problèmes et attaques confrontés par les systèmes de confiance

Dans [40] les auteurs proposent une description des problèmes fondamentaux et des attaques auxquels devrait faire face tout système de confiance. Les problèmes impactent la performance du système. Certains de ces problèmes représentent aussi les défaillances sur lesquels peuvent jouer les utilisateurs malveillants afin d'attaquer le système et par conséquent impacter sa robustesse. Les auteurs considèrent chaque phase du fonctionnement de ces systèmes. Nous présentons dans ce qui les problèmes relatifs à chaque phase et les attaques que nous avons choisi de traiter dans la thèse.

2.1.2.1 Problèmes confrontés par les systèmes de confiance

L'évaluation de la confiance dans un système de confiance comporte trois phases (voir Figure 2.1) :

La génération des évaluations : cette étape concerne la création des évaluations par les utilisateurs. Elle constitue une étape critique pour un système de confiance. Le premier enjeu auquel cette étape fait face est le besoin dans certains cas de transformer une évaluation qualitative basée sur l'avis de l'utilisateur en une valeur quantitative. Le deuxième enjeu est la présence d'utilisateurs malveillants dans le réseau. Parmi les problèmes de cette phase, nous citons : le manque de motivation pour produire des évaluations, la favorisation des évaluations positives, le problème d'initialisation, la possibilité de créer des identités à coût réduit ou nul, etc.

La distribution des évaluations : cette étape concerne l'architecture adoptée pour la distribution des évaluations dans le système.

FIGURE 2.1 – Système d'évaluation de la confiance

L'enjeu est de faire parvenir les évaluations à ceux qui ont besoin de prendre des décisions d'une manière correcte. Parmi les problèmes qui se posent pendant cette phase, nous citons : le manque de portabilité entre les systèmes et la latence de la valeur de la confiance, etc.

Agrégation des évaluations : cette étape concerne l'agrégation des évaluations afin de calculer la valeur de confiance. L'enjeu principal dans cette étape est de fournir une valeur de confiance utile pour un futur utilisateur de la ressource. De multiples problèmes peuvent intervenir pendant cette phase tels que : les équations incorrectes, la mémoire limitée, la sensibilité temporelle, etc.

2.1.2.2 Attaques contre les systèmes de confiance

Dans notre approche, nous avons choisi de se focaliser sur deux types d'attaques : les évaluations fausses/biaisées et l'attaque Sybil. Nous définissons ces attaques et les solutions qui ont été proposées pour les contourner. Ces deux types d'attaques concernent la phase de la génération de l'évaluation.

a) **L'attaque des évaluations biaisées :** il s'agit des évaluations émises par des utilisateurs mais qui ne correspondent pas à leur avis réel sur l'objet à évaluer. De telles évaluations, lorsqu'elles sont présentes dans un système d'évaluation de la confiance, peuvent fausser le calcul de la confiance. Elles peuvent être classées de la manière suivante :

i) *Evaluations malhonnêtes et injustes* : il s'agit d'évaluations faussement émises pour des motifs malhonnêtes. Cette attaque nous intéresse car elle est devenue très récurrente dans le Web. Elle s'est d'avantage amplifiée avec l'ouverture de compte à l'anonymat et l'absence ou le faible coût de soumission de commentaires/évaluations en ligne. Les évaluations malhonnêtes sont généralement identifiées en évaluant la crédibilité de l'utilisateur (ex., [26, 29]) ou à travers des calculs statistiques.

ii) *Collusion* : elle se produit lorsque deux ou plusieurs utilisateurs se mettent d'accord pour augmenter ou rétrograder la réputation d'un objet ou d'un autre utilisateur dans le réseau. Dans [5], les auteurs identifient trois types de collusion : le bourrage d'urne, le Bad Mouthing et la discrimination positive ou négative. Le premier ait lieu lorsqu'un utilisateur mène des fausses transactions et à l'issue de chacune de ces transactions, il publie un avis positif sur le fournisseur. Par conséquent, si cette attaque est faite à plusieurs reprises, la réputation du fournisseur est améliorée artificiellement. Le deuxième se produit lorsqu'un ou plusieurs utilisateurs conspirent contre un autre utilisateur dans la communauté afin de rétrograder sa réputation en leur attribuant des avis négatifs. Le troisième consiste à adopter un comportement discriminatoire aussi bien lors de l'offre d'une ressource ou lors de la génération des évaluations.

b) **L'attaque Sybil :** cette attaque est très occurrente dans les environnements en ligne où il est possible de créer des nouvelles identités à un coût minimal. Cette attaque peut se manifester sous deux formes :

i) *Collusion basée sur des identités multiples* : dans cette attaque des utilisateurs malveillants créent des entités virtuelles afin de soumettre de fausses évaluations. Deux types

d'approches ont été proposés pour lutter contre cette attaque. Dans le premier, l'architecture est centralisée. Une entité centrale est créée afin de vérifier l'exactitude et l'unicité des informations d'identification renseignées par l'utilisateur. Cette entité utilise souvent un modèle de paiement de crédits afin d'augmenter le coût d'obtention d'une identité. Dans le deuxième l'architecture est décentralisée. Certaines approches décentralisées imposent un identifiant unique pour chaque identité tel que l'adresse IP ou une clé publique (ex., le numéro de la carte d'identité nationale). D'autres approches utilisent les informations recueillis sur les réseaux sociaux et se basent sur les relations de confiance entre les utilisateurs afin d'identifier les plus fiables [42, 52].

ii) *Réinscription* : il s'agit d'une stratégie dans laquelle un utilisateur malveillant dont l'identité a été dévoilée, disparaît et se réinscrit au système sans payer de frais (ex., les sites commerciaux eBay et Amazon). Les systèmes résilients à ce type d'attaques sont ceux pénalisant les utilisateurs récemment inscrits [53].

2.2 Analyse des approches de confiance

Plusieurs approches d'évaluation de la confiance ont été définies et implémentées dans divers domaines (ex. e-commerce, les environnements orientés services etc). Plusieurs études ont alors été proposées afin de classifier les systèmes de confiance et de les comparer entre eux.

Dans l'étude [40] réalisée par Tavakolifard et al., les auteurs classifient les systèmes de confiance existants selon plusieurs dimensions (voir Figure 2.2). Cette classification représente une base solide pour la compréhension de l'état de l'art actuel des systèmes de confiance et donne une vue d'ensemble sur les thématiques de

recherches existantes et nous permet ainsi d'identifier celles qui nécessitent des travaux plus poussés. La classification sert à mettre en évidence les différences entre les différents systèmes et les comparent selon les différentes dimensions prédéfinies.
Afin de mieux nous positionner par rapport aux travaux existants, nous allons définir ces dimensions et préciser où nous nous situons par rapport à chaque dimension de cette taxonomie.

a) **La source de l'information :** deux types d'informations sont considérés dans les systèmes de confiance existants pour calculer la confiance d'une ressource : les expériences personnelles ou les recommandations des autres utilisateurs.
   Dans cette thèse, nous utilisons les recommandations des utilisateurs exprimées sous forme d'évaluations. Ces dernières sont particulièrement utiles lorsqu'il y a un manque d'expériences personnelles.

b) **L'architecture :** deux types d'architectures sont possibles : centralisée et distribuée. Dans la première, une autorité/entité centrale collecte les évaluations de la part de tous les membres de la communauté. A partir de ces évaluations, l'entité dérive une valeur de réputation publique et la partage avec toute la communauté. Dans la seconde, les avis sont fournis sur demande de l'utilisateur. Ce dernier calcule sa propre valeur de confiance sur la base des évaluations reçues.
   Nous optons dans cette thèse pour une architecture distribuée. En effet, l'architecture centralisée est plus vulnérables aux attaques, car il suffit d'attaquer l'unité centrale et comprendre son fonctionnement pour attaquer le système.

c) **Le type de l'information :** il s'agit du type d'information choisie pour l'évaluation de la confiance : explicite ou implicite. Dans cette thèse, nous utilisons les deux types d'information. La première est sous forme d'évaluations collectées auprès des utilisateurs. La deuxième est extraite à partir de l'analyse de la topologie du réseau social et les relations d'amitiés

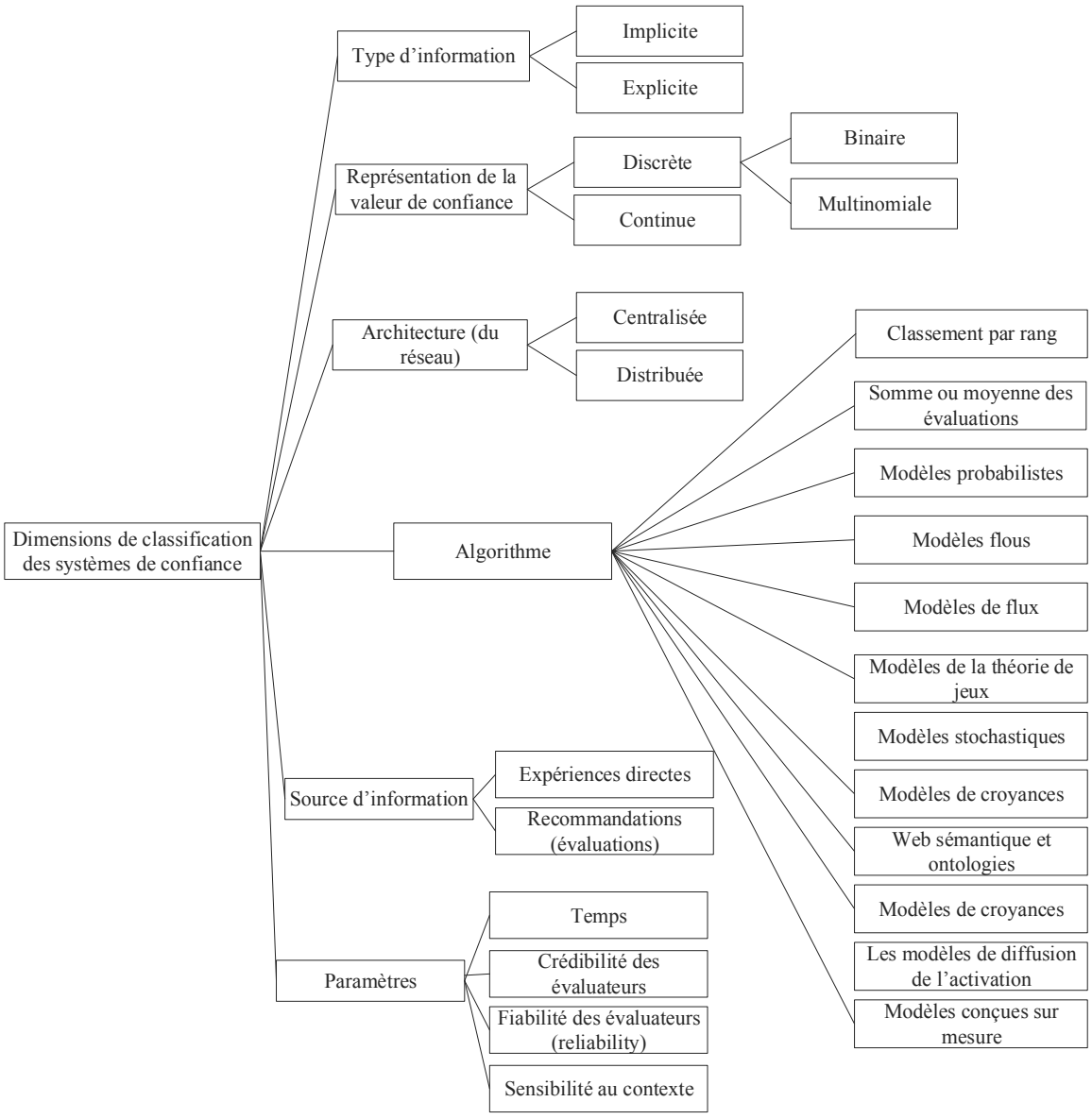


FIGURE 2.2 – Taxonomie des dimensions de classification des systèmes de confiance

entre les utilisateurs afin de démystifier les utilisateurs Sybil (CF. chapitre 4). Ce type d'analyse est aussi utilisé dans les réseaux sociaux comme Facebook et Linkedin afin d'évaluer la réputation d'un utilisateur dans une communauté donnée [31].

d) **L'algorithme :** il s'agit de la méthode de calcul de la valeur de confiance. Parmi ces méthodes, les plus proches de notre travail sont l'agrégation et les approches probabilistes. L'agrégation consiste à réaliser une nous utilisons une approche similaire aux travaux cités dans [26] et [29] et qui permettent de prendre en compte la crédibilité de l'utilisateur lors du calcul de la confiance (CF. Chapitre 3). Les approches probabilistes évaluent la confiance en mettant à jour une densité de probabilité (PDFs). Dans cette thèse, nous proposons une nouvelle approche utilisant également la probabilité mais à travers les bases de données probabilistes afin de lever l'incertitude liée à la cohérence des évaluations (CF. chapitre 5).

e) **La représentation de la valeur de confiance :** la valeur de la confiance est soit continue soit discrète. Les valeurs discrètes et continues diffèrent d'un modèle à un autre. Les valeurs discrètes peuvent être comprises entre un intervalle donné ou même incluant l'infinité. De plus, la valeur discrète peut être soit binaire soit binomiale. La représentation binaire permet de représenter uniquement une confiance absolue en la ressource ou pas de confiance du tout. La valeur binomiale permet de représenter des situations ou le degré de confiance n'est pas complet mais suffisant pour la situation en cause. Les valeurs continues sont représentées par des nombres réels souvent modélisés par des valeurs de probabilité subjective ou objective.

f) **Les paramètres** : il s'agit de paramètres critiques permettant l'amélioration de la valeur de confiance. A titre d'exemple, nous citons la prise en compte du temps de sorte que les informations les plus récentes aient plus d'influence sur le calcul

de la confiance, la crédibilité des utilisateurs et la sensibilité du système au contexte, etc. Dans cette thèse, nous prenons en compte le paramètre de crédibilité de l'utilisateur lors de l'évaluation de la confiance en considérant une définition plus complète de la crédibilité comportant sa fiabilité et son expertise.

2.3 Approches à base de crédibilité

Ces approches calculent la crédibilité de l'utilisateur en utilisant des algorithmes de clustering tel que k-means [26], des mesures de similarité [29, 48] ou des modèles mathématiques [45, 46].

2.3.1 Approches basées sur le filtrage

Dans [24], Azaza et al. proposent une approche de filtrage des avis collectées en ligne. L'approche comporte deux étapes. La première consiste à détecter les utilisateurs Sybil en utilisant des techniques qui combinent plusieurs critères tels que le profil des utilisateurs et la date de publication des avis. La deuxième étape évalue la crédibilité des utilisateurs à travers un modèle basé sur la consistance des avis et de l'influence qu'exercent les utilisateurs les uns sur les autres. Le module de détection des utilisateurs Sybil combine plusieurs types de comparaison des identifiants des utilisateurs : comparaison d'adresses e-mail, de nom de profils, de dates de publications d'avis et de listes d'amis. Chaque critère est utilisé individuellement et en résulte des classifications différentes de l'ensemble des utilisateurs selon le critère choisi. La technique de Classification Ascendante Hiérarchique (CAH) est utilisée par la suite afin d'arriver à une classification unique des utilisateurs. Chaque classe correspond à un ensemble d'utilisateurs Sybil créés probablement par le même utilisateur.
Dans [33], Riggs et al. associent la crédibilité d'un utilisateur à

son niveau d'expertise. Les auteurs proposent un algorithme de filtrage collaboratif pour l'évaluation automatique de l'expertise des examinateurs des productions scientifiques publiées gratuitement. Le but est de fournir un système plus flexible que le système académique suivi dans les conférences ou les journaux scientifiques reconnus. L'hypothèse de ce travail, est que les lecteurs seront plus intéressés par les articles révisés par les meilleurs examinateurs (experts) que ceux ayant les meilleures évaluations. La notion de consensus a été aussi utilisée. Néanmoins, le consensus est représenté par la moyenne des évaluations et non par l'avis de la majorité. Plus l'évaluation d'un examinateur est proche de la moyenne des évaluations plus son niveau d'expertise est élevé et par conséquent sa crédibilité.

Dans [21], Kim et al. se sont basées l'algorithme de filtrage collaboratif proposé par Riggs et al. [33] pour évaluer la crédibilité des évaluateurs. Les auteurs utilisent les jeux de données du site Epinion.com pour calculer la confiance basée sur l'expertise des utilisateurs dans une catégorie de produits (ex., films, livres, etc). L'expertise d'un utilisateur est établie en fonction de la qualité de ses évaluations dans cette catégorie.

2.3.2 Approches basées sur des mesures de similarité

Dans [48], Xiong et al. ont développé Peertrust, un outil d'évaluation de la confiance basé sur la crédibilité dans le contexte des réseaux pair-à-air. L'évaluation d'un pair se réfère à la satisfaction d'un autre suite à sa participation dans des opérations communes. Un pair peut avoir des intentions malicieuses en émettant de fausses évaluations. Les auteurs proposent d'attribuer un poids aux évaluateurs selon leur degré de crédibilité. La crédibilité d'un pair (p_i) est calculée en fonction de deux métriques : la qualité de service fourni par p_i et la similarité entre les évaluations d'un utilisateur et celles des autres évaluateurs.

2.3.3 Approches basées sur des modèles mathématiques

Dans [46], Whitby et al., les auteurs se sont intéressés à un type particulier d'évaluations dites biaisées. Les auteurs partent de l'observation que ces évaluations biaisées suivent un pattern statistique différent de celles non-biaisées. Ils proposent une technique de filtrage basée sur la distribution Bêta comme pattern statistique de référence. Cette technique distingue les évaluations qui se trouvent dans la région de rejet délimitée par la distribution Bêta des autres évaluations représentant alors la majorité. Cette technique est néanmoins inefficace lorsque la majorité des évaluateurs sont malhonnêtes.
Dans [45], Weng et al., se sont intéressés aux fausses évaluations dans les systèmes bayésiens d'évaluation en ligne. Ces systèmes tracent le comportement des acheteurs lors des transactions passées afin de prédire les transactions futures. Pour évaluer la qualité des évaluations, les auteurs utilisent la notion d'entropie comme une mesure de l'incertitude de l'information [6]. L'entropie permet d'exclure une évaluation fournie par un acheteur si cette dernière améliore ou dégrade considérablement la qualité de l'avis de la majorité.

2.3.4 Discussion

Les approches de confiance à base de crédibilité que nous venons de présenter supposent que les utilisateurs possèdent une bonne expertise et/ou fiabilité. Cependant, ces approches négligent les utilisateurs possédant à la fois une bonne expertise et fiabilité. Nous appelons ces utilisateurs des *utilisateurs stricts* (ou *sévères*). Ces derniers n'ont aucun intérêt à s'aligner avec la majorité. Pour atteindre un consensus, nous démontrons dans le chapitre 3 comment la technique de clustering flou réduirait l'écart entre les évaluations des "*utilisateurs stricts*" et l'avis de la majorité actuel. Dans cette

technique l'incertitude est représentée par des frontières graduelles entre les groupes d'avis à la place de frontières nettes entre eux. L'incertitude s'exprime par un degré d'appartenance d'une évaluation à un ou plusieurs groupes. Ainsi, les évaluations se trouvant à la frontière du groupe majoritaire et pouvant correspondre à des évaluations des utilisateurs stricts seront incluses dans ce dernier.

2.4 Approches résilientes aux attaques Sybil

Les attaques Sybil sont largement traitées dans la littérature. Certaines solutions se concentrent sur comment diminuer l'impact des attaques réalisées par ces utilisateurs. D'autres solutions préconisent la détection des utilisateurs Sybil en analysant les profils des utilisateurs. Nous présentons d'abord les travaux les plus pertinents de la littérature.

2.4.1 Approches à base de réseaux sociaux

Dans [8] Danezis et al. proposent SybilInfer, un algorithme étiquetait les noeuds dans un réseau social par honnête ou Sybil. Les auteurs définissent un modèle probabiliste de réseaux d'utilisateurs honnêtes pour localiser des régions potentielles d'utilisateurs malhonnêtes. De même Mislove et al. proposent Ostra, un algorithme utilisant les relations de confiance telles que les liens sociaux afin d'éviter toute communication indésirable entre les noeuds honnêtes et les noeuds malhonnêtes [28]. Les deux algorithmes SybilInfer et Ostra supposent une connaissance globale du réseau social tel que sa structure et les profils des nœuds. Cependant, Ostra ne garantit pas d'écarter correctement les noeuds Sybil.
Dans [14] Hota et al. présentent deux algorithmes pour la détection des noeuds Sybil : de routage multi-chemins et de vérification. Le premier algorithme recherche tout segment commun dans un chemin reliant un noeud dit vérificateur au groupe de noeuds soup-

çonnés d'être Sybil. Le deuxième sert à confirmer le statut de ce groupe en sélectionnant d'une manière aléatoire quelques noeuds et les sondant. Ces noeuds doivent répondre dans un délai assez court. Si une entité possède plusieurs identités, elle ne pourra pas répondre dans ce délai.

2.4.2 Discussion

Pour conclure cette partie de l'état de l'art, nous remarquons que la sélection aléatoire (ex., la recherche en largeur à l'aveugle) des utilisateurs met en jeu le nombre et la qualité des évaluations considérées lors du calcul de la confiance d'un SW. À cet effet, nous préconisons la recherche heuristique pour guider la collecte des évaluations vers les utilisateurs les plus appropriés aussi bien en termes de quantité que de qualité. Cela devrait garantir une meilleure qualité de la valeur de la confiance calculée.

2.5 Approches probabilistes

Dans les approches probabilistes existantes de gestion de confiance (ex. [41], [54], et [50]) les utilisateurs s'appuient soit sur leur propre expérience avec les ressources, soit sur les évaluations fournies par d'autres paires. Les évaluations fausses sont alors traitées par un mécanisme de filtrage approprié. Par la suite, nous décrivons les trois approches probabilistes existantes.
Dans [41] Teacy et al. proposent TAVOS un modèle de confiance pour les systèmes agents. Dans ce modèle un agent/pair gagne la confiance d'un autre au fil des interactions directes avec ce dernier. L'évaluation du résultat issu des interactions est binaire : réussite ou échec. Une fonction de densité modélise la probabilité d'occurrence d'une interaction réussie avec un pair. Le modèle se base sur les expériences des autres pairs pour calculer la confiance lorsque le pair n'a pas d'expérience directe. Le modèle utilise la crédibi-

lité des agents pour filtrer les évaluations incorrectes provenant d'agents ayant des connaissances limitées ou un comportement malveillant.
Dans [54] Zhou et al., proposent PowerTrust un système de confiance pour les réseaux pair-à-pair. Les nœuds évaluent les interactions avec les autres pairs et estiment la confiance localement en utilisant une technique d'apprentissage *bayésien*. La valeur globale de la confiance est ensuite calculée en utilisant ces valeurs locales. Cette valeur est mise à jour périodiquement en utilisant l'algorithme de marche aléatoire (Look-ahead Random Walk [25]) dans le système. L'algorithme sélectionne les pairs avec des valeurs locales de confiance au-delà d'un seuil prédéfini pour mettre à jour la valeur globale de confiance d'un pair donné. Cet algorithme utilise une table de hachage distribuée et indexée par des valeurs locales de confiance pour classer les pairs.
Dans [50], Yu et al. proposent un modèle probabiliste de la confiance basé sur la théorie de Dempster-Shaker [37]. Cette théorie permet de combiner des preuves provenant de plusieurs sources afin de calculer la probabilité d'occurrence d'un événement. Le modèle proposé étend la théorie des probabilités avec la modélisation de l'incertitude. Comme la somme des probabilités des résultats n'est pas nécessairement égale à 1, la probabilité résiduelle est considérée comme étant un état d'incertitude. Le modèle propose deux types de croyance : l'agent A croit que l'interaction avec l'agent B sera une réussite et que l'agent B ne remplira pas ses engagements. Le modèle utilise a priori les expériences directes pour calculer la confiance.

2.5.1 Discussion

En résumé, nous remarquons que les approches probabilistes mentionnées ci-dessus négligent l'interprétation des évaluations en présence de l'incertitude. En effet, chaque évaluation est vraie à un certain degré et fausse à un autre degré. Nous modélisons alors les

évaluations de l'utilisateur par une base de données probabiliste, nous interprétons les évaluations en termes de mondes possibles et nous calculons la confiance d'une ressource sur le Web comme une évaluation de requête sur une base de données probabiliste. Nous présentons plus en détails cette approche dans le chapitre intitulé le modèle de confiance.

2.6 Comparaison de notre approche avec certaines approches existantes

D'après l'état de l'art que nous avons présenté, nous pouvons constater que la recherche dans le domaine de la confiance est assez fructueuse. Nous disposons alors d'une base riche de travaux existants. Notre approche constitue alors une continuité de certains d'entre eux. Nous présentons dans cette section les approches d'évaluation de la confiance nous ayant le plus inspiré dans cette thèse. Certes, leur présentation facilitera davantage la compréhension de notre approche. Les deux premiers sont RateWeb [26] et CloudArmor [29] deux systèmes d'évaluation de la confiance à base de crédibilité. Le troisième est SumUp [42] un système d'agrégation de votes sur du continu en ligne (sur le Web). Même si les auteurs n'emploient pas le mot confiance, l'agrégation des votes représente une évaluation de la fiabilité de l'objet en question et par conséquent peut être assimilée à la confiance de ce dernier. Dans cette section, en premier, nous décrivons en détails les trois systèmes RateWeb, CloudArmor et SumUp. En second lieu, nous présentons un comparatif de notre approche avec ces derniers.

2.6.1 RateWeb

Dans [26], Malik et al. proposent RateWeb est un système d'évaluation de la réputation pour les environnements orientés service

qui a pour but d'optimiser la sélection et la composition des services Web.

2.6.1.1 Fondements

Dans RateWeb, la qualité de service (QoS) du service Web est perçue comme un indicateur de la confiance que les consommateurs peuvent lui accorder. Elle est définie comme "a set of quantitative and qualitative characteristics of a system necessary to achieve the required functionality of an application". Il s'agit alors d'un mapping entre un ensemble de paramètres qualité et un ensemble de valeurs ou des intervalles de valeurs. Par exemple : le temps de réponse, le prix d'invocation, la disponibilité, l'accessibilité, etc. Chaque consommateur de services enregistre sa propre perception sur la confiance de seulement les services qu'il invoque. Cette perception est considérée comme son évaluation personnelle. L'agrégation de toutes les évaluations personnelles d'un service Web donné afin de dériver une valeur unique est définie comme la confiance de ce dernier. Malik et Bouguettaya analysent les évaluations selon leur distance par rapport à l'avis de la majorité. L'avis de la majorité est obtenu par l'application de l'algorithme de clustering $\mathcal{K}$-means regroupant des évaluations similaires dans des groupes. Le groupe le plus fortement peuplé est désigné comme groupe majoritaire et son centroîde représente l'avis de la majorité. RATEWeb est conçu conformément aux méthodologies des réseaux sociaux de la vie réelle. Il évalue alors la réputation en utilisant un certain nombre de métriques afin de prendre en compte l'aspect dynamique de ces environnements et également la possibilité de présence d'utilisateurs malveillants dont le comportement varie entre honnête et malhonnête dans leur évaluation des services. Les métriques de RATEWeb sont définies pour capturer la plupart des aspects de la confiance d'un point de vie social. Ces métriques sont : l'historique des évaluations de l'utilisateur, l'évaluation personnalisée de la confiance par les préférences personnelles de l'uti-

lisateur, l'utilité des évaluations fournies par l'utilisateur et la sensibilité temporelle.

2.6.1.2 Algorithme

La figure 2.3 est une représentation graphique du fonctionnement de l'algorithme d'évaluation de réputation selon les différentes métriques définies ci-dessus. L'entrée à l'algorithme est une liste de consommateurs ayant interagi avec le/les fournisseur(s) de services et ont ainsi une évaluation de la confiance de ces derniers. L'algorithme itère sur une liste complète de fournisseurs potentiels à partir d'un registre UDDI (Universal Description, Discovery, and Integration). La sortie de l'algorithme est le fournisseur de services ayant la réputation la plus élevée. L'algorithme itère ensuite sur une liste de consommateurs de service ayant déjà interagi avec le fournisseur du service en question s_j et collecte les évaluations de la réputation. Chaque consommateur de service renvoie un vecteur RSV composé d'évaluations de la réputation pour chaque attribut de qualité ainsi que la date d'évaluation de cette réputation. A ce moment, le consommateur compare le RSV de chaque consommateur avec le sien. Si les valeurs sont similaires, l'évaluation est acceptée. Dans le cas ou les deux RSV sont différents une nouvelle évaluation est calculée selon les préférences personnelles du consommateur. Toutes les évaluations sont utilisées pour calculer l'avis majoritaire suivi. Ensuite, la crédibilité des consommateurs du service C_r est calculée sur la base de l'avis majoritaire, la dernière valeur de réputation calculée et l'évaluation personnelle du consommateur. La crédibilité est par la suite ajustée par le facteur d'utilité U_f. Une moyenne pondérée des réputations collectées V_i par les valeurs de crédibilité des évaluateurs Cr_i est ensuite calculée. Cette moyenne est diluée par la valeur de f_d pour former la valeur de réputation finale du service s_j. Si cette valeur est supérieure à la valeur de réputation du dernier fournisseur dans la boucle, alors s_j est marqué comme le service ayant la meilleur va-

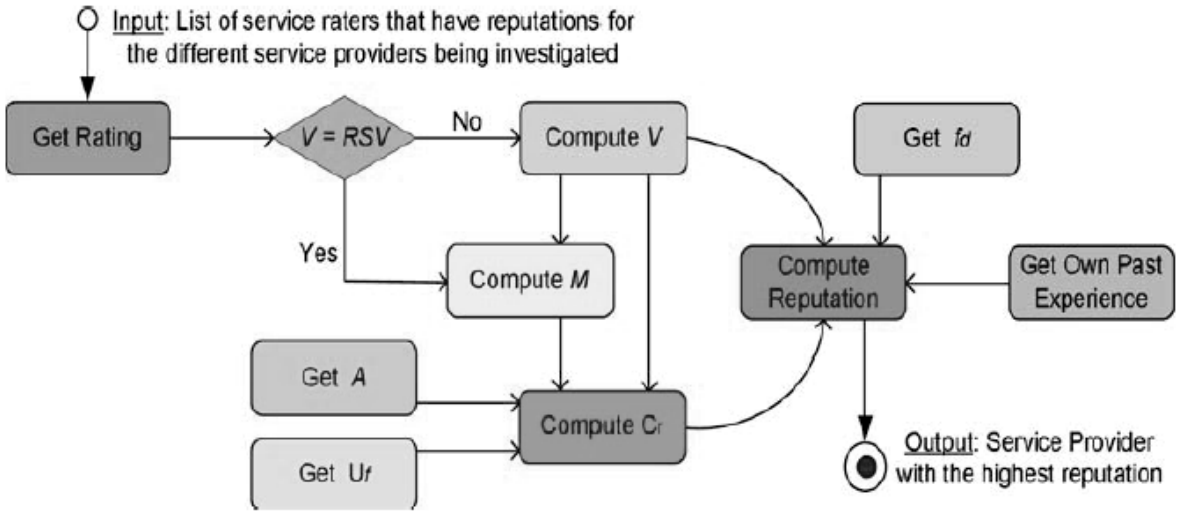


FIGURE 2.3 – Métriques d'évaluation de la réputation dans le système RateWeb

leur de réputation.

2.6.2 CloudArmor

Dans [29], Noor et al. proposent CloudArmor est un système d'évaluation de la confiance dans le contexte du Cloud Computing. La caractéristique principale du système est la protection de la vie privée de l'utilisateur en préservant l'anonymat de leur identité et leurs interactions.

2.6.2.1 Fondements

Le système est basé sur deux modèles : un modèle de crédibilité des évaluations protégeant le système des utilisateurs malveillants et préservant la vie privée des utilisateurs et un modèle de disponibilité pour gérer la disponibilité de l'implémentation décentralisée du système.
Dans le premier modèle, la crédibilité est définie en fonction de deux facteurs : le consensus majoritaire et la densité des évaluations. Pour mesurer la distance entre l'évaluation de l'utilisateur et l'avis de la majorité, les auteurs utilisent la moyenne quadra-

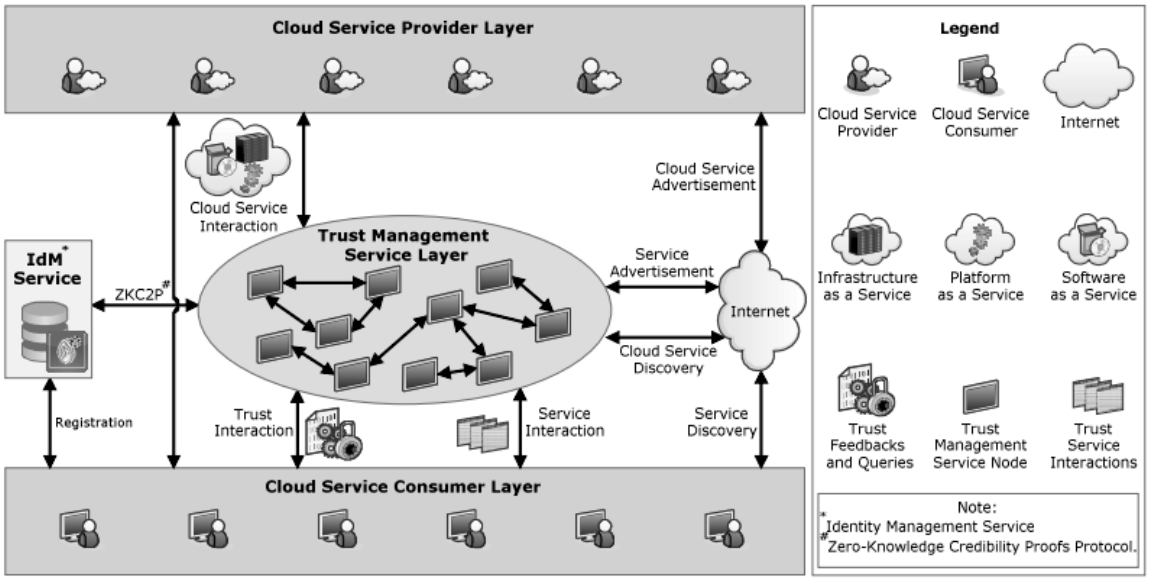


FIGURE 2.4 – Système d'évaluation de la confiance CloudArmor

tique. La densité des évaluations dans ce travail permet de détecter les fausses évaluations fournies par des utilisateurs évaluant plusieurs fois un service Cloud pendant une courte période. En outre, le modèle de crédibilité utilise des métriques pour la détection des attaques Sybil fournissant des fausses évaluations, y compris la reconnaissance des utilisateurs ayant des identités multiples et les attaques Sybil occasionnelles.
Le modèle de disponibilité assure la diffusion de plusieurs noeuds d'une manière distribuée pour qu'ils se chargent des échanges des évaluations entre les utilisateurs d'une manière centralisée. Le modèle exploite alors des techniques de répartition de charge (en anglais, load balancing) pour le partage de la charge de travail assurant ainsi le niveau de disponibilité souhaité du système.

2.6.2.2 Architecture

Le système CloudArmor est essentiellement fondé sur l'architecture orientée services (calque de l'anglais service oriented architecture, SOA) dans le sens où les ressources, par exemple, les in-

frastructures, les plates-formes et logiciels, sont exposés dans des nuages (Clouds) comme des services. En particulier, le service de gestion de la confiance maintient plusieurs noeuds distribués qui fournirent des interfaces afin que les utilisateurs puissent donner des évaluations à leurs pairs ou leur demander les résultats de la confiance. La figure 2.4 est une illustration du framework CloudArmor, qui se compose de trois couches différentes, à savoir la couche du fournisseur du service Cloud (Service Cloud Provider), celle de la gestion de la confiance (Service Layer Management Trust), et la couche du consommateur du service Cloud (Cloud Service Consumer Layer).

Cloud Service Provider Layer. Cette couche se compose de différents fournisseurs de services Cloud offrant un ou plusieurs services Cloud, à savoir, IaaS (Infrastructure as a Service), PaaS (Platform as a Service) et SaaS (Software as a Service), publiquement disponible sur le Web. Ces services Cloud sont accessibles via des portails Web et indexés sur des moteurs de recherche tels que Google, Yahoo et Baidu. Les interactions possibles de cette couche sont celles entre les services Cloud avec les utilisateurs des services Cloud, le système de gestion de la confiance et les annonces des services Cloud où les fournisseurs sont en mesure d'annoncer leurs services sur le Web.

The Trust Management Service Layer. Cette couche se compose de plusieurs nœuds distribués de gestion de la confiance hébergés dans différents environnements Cloud dans différentes zones géographiques. Ces nœuds exposent des interfaces permettant aux utilisateurs de donner leurs avis ou demander les résultats de la confiance d'une manière décentralisée. Les interactions présentes dans cette couche comprennent : (i) les interactions avec les fournisseurs de services Cloud, (ii) l'annonce de services afin d'annoncer la confiance comme étant un service disponible sur Internet, (iii) la découverte de services Cloud pour permettre aux utilisateurs d'évaluer la confiance

des nouveaux services Cloud, et (iv) les interactions relatives à l'évaluation de la crédibilité des utilisateurs à travers le protocole ZKC2P (la crédibilité de Zero-Knowledge Protocole Proof) assurant la préservation de la vie privée des utilisateurs.

The Cloud Service Consumer Layer. Enfin, cette couche se compose de différents utilisateurs des services Cloud. Les interactions de cette couche sont : i) la découverte des services Cloud par les utilisateurs d'Internet, ii) les interactions de confiance et de services où les utilisateurs sont en mesure de donner leur avis ou récupérer les résultats de confiance d'un service Cloud particulier et iii) l'enregistrement des utilisateurs à travers le service d'identification (i.e, Identity Management Service). A travers ce service, les utilisateurs créent des identités en fournissant leurs informations d'identification avant d'avoir accès au système de gestion de la confiance.

2.6.3 SumUp

Dans [42], Tran et al. proposent SumUp conçu pour superviser les systèmes de vote/évaluation en ligne tels que Diggs, YouTube, eBay, Amazon, connus pour être sensibles aux attaques Sybil. Le système s'est montré efficace pour faire face aux attaques Sybil dans le monde réel, à travers son application sur la trace des votes du site Web `http://digg.com`.

2.6.3.1 Fondements

SumUp exploite le réseau social des utilisateurs créant des comptes sur ces réseaux afin de protéger les systèmes de vote en ligne contre les attaques Sybil. Le système affecte des capacités aux liens sociaux entre les utilisateurs, désignant la capacité de ces utilisateurs à émettre des évaluations. Cette capacité est d'autant plus élevée si les chances que l'utilisateur qu'il soit Sybil sont infimes. Par la suite le réseau est considéré comme un graphe sur lequel est appli-

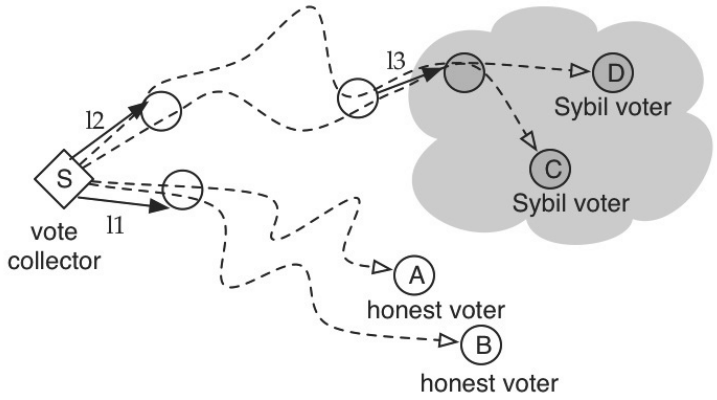


FIGURE 2.5 – Collecte de votes dans SumUp à partir d'un collecteur de vote aux votants A, B, C et D

qué l'algorithme de flot maximal afin de collecter les évaluations des utilisateurs. SumUp définit un arc d'attaque comme un lien social que l'utilisateur Sybil a réussi à établir avec un utilisateur honnête. Ainsi, SumUp parvient à limiter le nombre de faux votes (i.e, fausses évaluations) émis par les utilisateurs Sybil au nombre d'arcs d'attaques.

2.6.3.2 Algorithme

SumUp est basé sur l'algorithme de flot maximal. Ce dernier calcule un ensemble de chemins approximatifs à flot maximal dans le graphe de confiance à partir du collecteur de vote à tous les votants sur un objet donné. Le long de chaque lien traversé, le flot consomme une unité de la capacité du lien. La figure 2.5 montre un exemple des flots résultants du collecteurs aux votants A, B, C et D. Les lignes droites désignent les liens de confiance et les lignes pointillées représentent les chemins de flot de vote. Les chemins de flot de vote aux votants honnêtes sont congestionnés au niveau des liens proches du collecteur de vote alors que les chemins de flot de vote aux votants Sybil sont congestionnés mais au niveau des arcs d'attaques plus lointains.
Le calcul des flux de vote adaptatif utilise trois idées clés. Tout

d'abord, l'algorithme limite le nombre de votes collectés sur un objet à une valeur maximale C_{max}. Comme C_{max} est utilisé pour attribuer la capacité globale de vote dans le graphe de la confiance et selon l'observation précédente, plus ce nombre est petit par rapport au nombre total d'utilisateur moins l'attaquant a de capacité à voter. SumUp ajuste C_{max} de manière à collecter le plus grand nombre de votes honnêtes sur un objet donné.

Le deuxième aspect important dans SumUp concerne l'affection ou distribution des capacités aux liens de confiance entre les utilisateurs de manière à collecter un nombre maximal C_{max} (fixé a priori) de votes honnêtes et très peu de faux votes. Cette opération attribue à chaque utilisateur un nombre exact de tickets représentant sa capacité à voter. Tout d'abord, SumUp attribue C_{max} tickets au collecteur de vote. Ce dernier distribue d'une manière équitable les C_{max} tickets qu'il a reçu aux utilisateurs qui lui sont directement connectés à travers les liens de confiance. La distribution de tickets se poursuit à travers un parcours en largeur du réseau social considéré comme un graphe de confiance partant du collecteur de vote. Chaque utilisateur, distribue à son tour les tickets qu'il reçoit à ces fils dans le graphe de confiance. Comme le montre la figure 2.6, le processus de distribution de billets fait apparaître une enveloppe autour du collecteur de vote s ; au-delà de l'enveloppe tous les liens ont la capacité 1. L'enveloppe de vote contient C_{max} noeuds qui seront considérés comme points d'entrée pour l'algorithme de collecte de votes. Si C_{max} représente une estimation du nombre d'utilisateurs honnêtes dans le réseau social, alors l'enveloppe contient alors suffisamment de capacité pour collecter les C_{max} votes honnêtes. D'autre part, un arc d'attaque au-delà de l'enveloppe peut se propager au plus de 1 vote quel que soit le nombre de Sybil identités au-delà de cet arc.

L'idée clé finale de SumUp est d'exploiter les votes des utilisateurs afin de pénaliser ceux qui émettent des faux votes. Dans le contexte des attaques Sybil, la pénalisation des identités des utilisateurs est

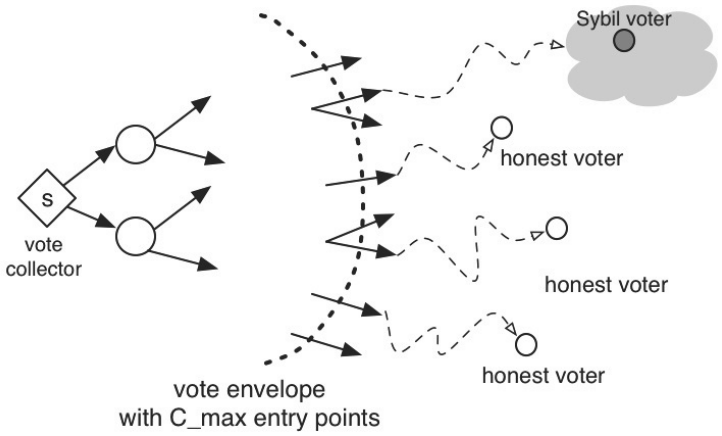


FIGURE 2.6 – Distribution des capacités d'évaluer aux utilisateurs dans SumUp

inefficace car ces derniers peuvent continuer à émettre des faux votes en utilisant de nouvelles identités Sybil. Cependant, comme un arc d'attaque est toujours présent dans le graphe de confiance, SumUp pénalise ce dernier en diminuant sa capacité.

2.6.4 Comparaison

Le tableau 2.1 compare notre approche aux autres approches selon les différentes dimensions décrites dans la section 2.2. Toutes les approches ont un but en commun qui est d'évaluer la confiance d'un certain type de ressource sur le Web : les services Web pour RateWeb et les services Cloud pour CloudArmor. De même que SumUp, nous présentons notre approche dans un contexte plus général vu qu'elle est applicable à l'évaluation de toute ressource sur le Web. Toutes les approches, utilisent des informations explicites sur les ressources uniquement sous forme de recommandations à l'exception du système RateWeb qui utilise aussi les expériences directes des consommateurs avec les services Web. En ce qui concerne l'architecture, nous avons opté pour une architecture décentralisée, tel est le cas de RateWeb et CloudArmor, à l'inverse de SumUp où une seule unité centrale se charge de la collecte des évaluations et du calcul de la confiance. Les approches utilisent

différents algorithmes pour le calcul de la confiance. La manière la plus évidente de calculer la confiance est de réaliser une moyenne des évaluations comme le fait le système SumUp. Les systèmes RateWeb et CloudArmor, calculent la confiance comme une moyenne pondérée par la crédibilité de l'utilisateur. Dans notre approche, nous proposons un algorithme innovateur de calcul de la confiance comme une évaluation de requêtes probabiliste afin de prendre en compte l'incertitude. Finalement, les différents paramètres et leur utilité seront présentés dans la description de chaque système.

Critère	Approches			
	RateWeb [26]	CloudArmor [29]	SumUp [42]	Notre approche
Type d'information	explicite	explicite	explicite	explicite
Source de l'information	expériences directes et recommandations	recommandations	recommandations	recommandations
Architecture	décentralisée	décentralisée	centralisée	décentralisée
Algorithme	moyenne pondérée	moyenne pondérée	moyenne des évaluations	moyenne pondérée et évaluation de requêtes probabilistes
Représentation de la valeur de la confiance	continue	continue	binaire	continue
Les paramètres	crédibilité, avis majoritaire, historique des évaluations, densité des évaluations préférences personnelles de l'utilisateur, expérience personnelle, sensibilité temporelle	crédibilité de l'utilisateur	fiabilité de l'utilisateur (son identité)	crédibilité, fiabilité de l'utilisateur

TABLEAU 2.1 – Comparaison des approches selon plusieurs dimensions

2.7 Conclusion

Afin de permettre la mise en place d'un système efficace et robuste d'évaluation de la confiance des ressources sur le Web, nous avons adopté une approche qui prend en compte les trois phases d'évaluation de la confiance, à savoir : la génération de l'évaluation par l'utilisateur, la distribution des évaluations dans le système et enfin l'agrégation des évaluations.

Il fallait tout d'abord définir les concepts fondamentaux liés au domaine de la confiance. Nous avons présenté un ensemble de définitions des concepts de base afin de lever toute ambiguïté sur le sens des termes utilisés et nous avons défini les différents problèmes que rencontrent les systèmes de confiance et les attaques qu'ils subissent.

La littérature regorge de système de confiance. Afin de mieux comprendre ces systèmes et d'identifier leurs principales différences, nous nous sommes basés sur des études existantes présentant des analyses pertinentes. Ces études ont relevé une taxonomie des dimensions de classification des systèmes de confiance. Cette taxonomie nous a permis de mieux comprendre l'état de l'art actuel et nous a dirigés vers les thématiques de recherches qui nous ont le plus inspirées. A l'issue de cette analyse, nous avons recensé les systèmes de confiance connexes à notre travail. En premier lieu, nous avons étudié les approches à base de crédibilité et résilientes aux attaques des fausses évaluations. Nous avons identifié trois types d'approches à base de crédibilité, celles basées sur le clustering ou le filtrage, celles basées sur les mesures de similarités et celles basées sur des modèles mathématiques. En second lieu, nous avons étudié les approches résilientes aux attaques Sybil. Enfin, comme nous proposons une approche d'évaluation de la confiance en utilisant les bases de données probabilistes, il était également nécessaire d'étudier les approches probabilistes présentes dans la littérature.

Avant d'aborder plus en détail l'approche que nous proposons, nous avons présenté à la fin de ce chapitre une comparaison de notre approche avec les approches les plus proches de la nôtre selon plusieurs dimensions.

Chapitre 3

Modèle de crédibilité flou résilient aux évaluations biaisées

Sommaire

3.1 Introduction

L'évaluation de la confiance des ressources sur le Web est confrontée au problème d'incohérence des évaluations. Afin d'obtenir une évaluation correcte de la confiance d'une ressource sur le Web, il est nécessaire de lever l'incertitude sur les évaluations causée par ce problème. Cette incohérence peut avoir une raison légitime comme une mauvaise expérience qu'a eue l'utilisateur avec la ressource dans un contexte particulier. Cependant, dans la plupart des cas, elle est liée à l'attaque des évaluations biaisées. Il s'agit d'une attaque réalisée par un utilisateur malhonnête émettant une fausse évaluation afin de promouvoir une ressource et rétrograder une autre. Plusieurs approches à base de crédibilité ont été proposées pour faire face à ce type d'attaques [26, 29].

La crédibilité d'un utilisateur est définie par deux composantes principales qui sont : l'expertise et la fiabilité. Lorsque les utilisateurs sont en désaccord sur une certaine évaluation d'une ressource, ils établissent un consensus en utilisant l'avis de la majorité. Les utilisateurs proches de l'avis de la majorité sont plus crédibles que ceux se trouvant éloignés. Cependant, ces approches négligent les utilisateurs possédant à la fois une bonne expertise et fiabilité. Nous appelons ces utilisateurs des *stricts* (ou *sévères*). Ces derniers n'ont aucun intérêt à s'aligner avec la majorité pour des raisons telles que leur objectivité et leur précision dans l'évaluation. Notre but est de réduire l'écart entre l'avis des utilisateurs *stricts* et l'avis de la majorité de telle sorte à prendre en compte leurs évaluations lors du calcul de la confiance.

Dans ce chapitre nous proposons un modèle de crédibilité résilient aux attaques des évaluations biaisées. Le modèle est basé sur une technique de clustering flou permettant de réduire l'écart entre les évaluations des utilisateurs strictes et celle de l'avis de la majorité.

Ce chapitre est organisé comme suit. Nous définissons tout d'abord les fondements de notre modèle de crédibilité. Ensuite nous pré-

sentons une description globale du modèle de crédibilité suivie par une description détaillée de chacun de ses composants : le clustering des évaluations, la recherche de l'avis de la majorité et le calcul de la crédibilité. Enfin, nous proposons une approche déterministe pour l'évaluation de la confiance combinant l'ensemble des évaluations des utilisateurs en fonction de leurs crédibilités.

3.2 Fondements du modèle de crédibilité

Dans cette section nous définissons les notions fondamentales de notre modèle de crédibilité à savoir le consensus majoritaire et les utilisateurs stricts, ainsi que la liaison entre ces deux notions.

3.2.1 Consensus majoritaire

Le consensus majoritaire ne représente pas un unique avis adopté par une majorité, mais plutôt l'apport de multiples avis différents. Il est atteint à travers l'adaptation progressive jusqu'à ce qu'une solution satisfaisant le plus grand nombre de personnes puisse être dégagée. Le consensus ne signifie pas forcément que tout le monde est satisfait du résultat, mais suggère plutôt que tout le monde peut juger le résultat acceptable et que la majorité est satisfaite.
Il est bien connu que la majorité sont en général d'accord avec les avis ou les jugements des experts au sujet de ce qui est bien et ce qui ne l'est pas. De même, nous partons de l'hypothèse que la majorité des utilisateurs des ressources sur le Web seront d'accord avec l'avis des experts. En d'autres termes, tout utilisateur dont l'avis est proche de l'avis de la majorité est considérée comme un utilisateur fiable et expert et par conséquent crédible. Néanmoins, il est possible qu'un utilisateur expert soit strict dans son jugement et modélisons dans le paragraphe suivant ce type particulier d'utilisateurs et comment ne pas les écarter lors de l'évaluation de la confiance d'une ressource.

3.2.2 Utilisateurs stricts

Les *utilisateurs stricts* sont caractérisés par une *solide* expertise et une *grande* fiabilité dans une certaine communauté. Ces utilisateurs maintiennent leurs avis indépendamment de l'avis de la majorité. Dans [35] les auteurs identifient plusieurs raisons pour lesquelles de tels utilisateurs ne changent pas d'avis. Parmi ces raisons, nous citons : véracité– ces utilisateurs disent la vérité, objectivité– leurs évaluations sont basées sur des évidences, et précision– ces utilisateurs estiment correctement leurs évaluations.

3.2.3 Liaison entre l'avis de la majorité et celui des utilisateurs stricts

Plusieurs études ont été menées en psychologie sociale (ex., [22] et [38]) pour évaluer l'impact de la crédibilité de la source sur les croyances et changements d'attitude. Ces études démontrent que les sources crédibles sont persuasives et peuvent influencer les croyances actuelles (ex. évaluations) et les attitudes des pairs de manière significative comparée aux sources non crédibles. Par conséquent, *les utilisateurs stricts*, en se basant sur leur expertise, peuvent encourager les utilisateurs à revoir leurs évaluations. Afin d'étudier ce phénomène, nous nous appuyons sur le paradigme d'*apprentissage actif* de Yager [49]. Les médias modernes font largement appel au paradigme de l'apprentissage actif afin de diffuser des messages publicitaires et vendre les produits commerciaux. De même, les politiciens s'appuient sur leur habilité à expliquer clairement afin de nous persuader de leurs idées et de ce qu'ils croient être corrects. Un autre exemple de la pratique du paradigme de l'apprentissage actif se trouve dans les salles de tribunal. En effet, l'interaction entre un avocat voulant défendre son client et le juge est basée sur sa maîtrise de la manière dont le juge traite et analyse l'information. Également un juge profite de sa compréhension du processus d'apprentissage humain pour expliquer des questions

juridiques complexes.
Dans notre cas, le paradigme d'apprentissage actif s'applique dans des situations où les évaluations sont correctes (resp. fausses) mais pas nécessairement précises, demandant un léger (resp. important) raffinement par les membres appartenant à la majorité. Notre proposition consiste donc à réduire l'écart entre les évaluations des utilisateurs stricts et celles de la majorité actuelle afin d'atteindre un consensus. Nous considérons que les utilisateurs stricts peuvent appartenir à des degrés différents à plusieurs groupes d'avis, et peuvent influencer les croyances de ces groupes de différentes manières (ex., fortement ou faiblement). Les appartenances *fortes* et *faibles* à un certain groupe d'avis (ou cluster) peuvent être incertaines et dépendent du domaine et/ou des préférences de l'utilisateur. En conséquence, lors de la recherche de l'avis de la majorité, nous avons opté pour une technique de clustering flou des évaluations afin de traiter l'incertitude de l'appartenance d'un avis à différents clusters d'avis. Cette technique nous permet d'inclure les avis des utilisateurs stricts dans l'avis de la majorité. Une fois l'avis de la majorité établie, nous calculons par la suite la crédibilité des utilisateurs selon leur écart par rapport à cette dernière.

3.3 Description globale du modèle de crédibilité

D'une manière générale, un modèle de crédibilité prend en entrée un ensemble d'évaluations émises par les utilisateurs et génère en

FIGURE 3.1 – Vue globale du modèle de crédibilité flou

sortie les crédibilités associées à chacun d'entre eux (voir Figure 3.1). Notre modèle de crédibilité comporte trois étapes. Premièrement, le clustering des évaluations qui prend en entrée le choix d'un l'algorithme de clustering donné. Ce dernier génère un ensemble de clusters ayant des limites rigides dans le cas du clustering classique ou floues dans le cas du clustering flou. Deuxièmement, la recherche du cluster majoritaire représentant l'avis de la majorité. Finalement, le calcul de la crédibilité des utilisateurs, en mesurant leur distance par rapport à l'avis de la majorité.

3.3.1 Modélisation des évaluations

Soit $\mathcal{L}$ l'ensemble de n utilisateurs $\sqcap_{i=1,n}$ ayant des avis sur la performance d'une ressource R. Notre modèle de crédibilité prend en entrée l'ensemble des évaluations $(\mathcal{X}_{i=1,n})$ fournies par $\mathcal{L}$. La sortie du modèle est une valeur de crédibilité $\mathcal{CR}_k$ associée à chaque utilisateur u_k.

3.3.2 Exemple d'application

Afin d'illustrer notre méthode, nous proposons un exemple d'évaluations d'une ressource $\mathcal{R}$ par 10 utilisateurs (Figure 3.1). Nous considérons que l'avis de la majorité est 0.86 et que parmi les 10 utilisateurs, il existe deux qui sont stricts notamment u_3 et u_4. Ces derniers fournissent une évaluation plus objective de la ressource compte tenu de leur expertise. Notre objectif est d'estimer la crédibilité de chacun de ces utilisateurs sur la base des évaluations qu'ils ont fournies. Nous souhaitons démontrer à travers cet exemple comment le clustering flou permet de réduire l'écart entre l'avis de la majorité et les utilisateurs stricts u_3 et u_4.

Utilisateur u_i	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
Évaluation X_i	$\mathcal{X}_1$=0.2	$\mathcal{X}_2$=0.2	$\mathcal{X}_3$=0.7	$\mathcal{X}_4$=0.71	$\mathcal{X}_5$=0.86	$\mathcal{X}_6$=0.86	$\mathcal{X}_7$=0.86	$\mathcal{X}_8$=0.86	$\mathcal{X}_9$=0.86	$\mathcal{X}_10$
Crédibilité $C\mathcal{R}_i$	?	?	?	?	?	?	?	?	?	?

TABLEAU 3.1 – Évaluation d'une ressource $\mathcal{R}$ par des utilisateurs

3.4 Clustering des évaluations

Le clustering est une technique d'apprentissage non supervisé ayant pour but de tirer des informations/connaissances sur des données ou de déterminer les liens entre ces derniers. Cette technique permet de structurer les données en classes homogènes (clusters) tels que les données d'un cluster soient le plus similaires possibles. Les techniques de clustering tels que $\mathcal{K}$-means [19] et fuzzy C-means [1] permettent de générer des clusters de manière robuste et d'éliminer aussi bien le bruit que les cas extrêmes.

3.4.1 Technique de clustering classique

Les approches actuelles de confiance basées sur la crédibilité telles que [26] et [29] utilisent l'algorithme de $\mathcal{K}$-means afin d'établir le consensus majoritaire. Afin de mieux motiver notre choix du clustering flou, nous expliquons dans ce qui suit le principe de cet algorithme et ses limites dans la recherche du cluster majoritaire.

3.4.1.1 Algorithme de clustering K-means

L'algorithme $\mathcal{K}$-means ou k moyennes est l'algorithme le plus populaire des méthodes de clustering classique. Il doit sa popularité à sa simplicité et sa capacité de traiter de larges ensembles de données. Ce dernier permet de partitionner un ensemble d'objets/points en K clusters concentrés et isolés les uns des autres. K étant un nombre fixé par l'utilisateur. Chaque cluster possède un centroide représentant son centre de gravité. Dans les approches actuelles de confiance à base de crédibilité, l'avis de la majorité

$\mathcal{M}_K$ est représenté par le centroïde du cluster le plus peuplé dit cluster majoritaire.

Cependant, cet algorithme n'est pas toujours approprié dans la recherche du cluster majoritaire. En effet, la définition des clusters, ne garantit pas la présence d'un unique cluster majoritaire. Si les avis sont controversés sur la qualité d'une ressource donnée (ex. autant d'avis positifs que d'avis négatifs), il est inconcevable de laisser au hasard le choix du cluster du cluster majoritaire. De plus, dans la plupart des cas, il existe une grande palette d'avis qui dépasse K, le nombre de clusters préalablement choisi. Un avis pas très positif peu par exemple être considéré comme étant négatif. Néanmoins, l'algorithme $\mathcal{K}$-means ne permet pas un recouvrement des clusters. Un objet ne peut appartenir qu'à un cluster à la fois.

En ce basant sur l'exemple de la sous-section 3.3.2 si l'on considère que l'évaluation correcte de la ressource R est 0.86. Un utilisateur ayant fourni une évaluation de la ressource appartient tout à fait à la classe des utilisateurs crédibles s'il fournit une évaluation supérieure à 0.86 et qu'il n'y appartient pas du tout s'il fournit une évaluation inférieure. Il est aberrant de considérer qu'un utilisateur ayant évalué R à 0.87 est crédible et un autre l'ayant évalué à 0.85 ne l'est pas du tout. Ceci peut être le cas d'un évaluateur strict exclu de l'avis de la majorité car son avis dévie légèrement.

Pour pallier à ces problèmes, nous proposons d'utiliser le clustering flou qui permet une représentation simple des incertitudes et des imprécisions liées aux données.

3.4.2 Technique de clustering flou

Dans cette sous section, nous décrivons d'abord les fondements du clustering flou et ensuite l'algorithme de clustering flou Fuzzy C-means que nous utilisons afin d'établir le consensus majoritaire.

3.4.2.1 Fondements

Le clustering flou est basé sur la notion de sous-classes floues (ou clusters flous) a pour but de permettre des graduations dans l'appartenance d'un élément à une classe, c'est-à-dire d'autoriser un élément à appartenir plus ou moins fortement à cette classe. Elle évite ainsi l'utilisation arbitraire de limites rigides à des classes présente dans le clustering classique où cette appartenance est binaire (appartient ou n'appartient pas à un ensemble).
La notion de sous-classes floues a été introduite en mathématiques par Zadeh en 1965. Elle a été ensuite développée par Kauffmann en se basant sur le fait que "*Most real-world classes are fuzzy in nature in that the transition from membership to non-membership in such classes is gradual rather than abrupt. This, given an object $\mathcal{X}$ and a class $\mathcal{F}$, the real question in most cases is not whether x is or not a member of F, but the degree to which x belong to F*" [20].
Cette notion permet l'utilisation de catégories aux limites mal définies (ex. adulte ou vieux), de situations intermédiaires entre le tout et le rien (ex. presque correct), le passage progressif d'une propriété à une autre (ex. passage d'une température de froid à tiède), l'utilisation de valeurs approximatives (environ 2 m). Si nous reprenons l'exemple précédent, un utilisateur ayant fourni une évaluation inférieur à 0.86 m peut appartenir à la classe des utilisateurs crédibles et plus son évaluation se rapproche de 0.86 plus son appartenance à cette classe est forte. Ainsi, dans notre cas, un utilisateur strict peut appartenir au cluster majoritaire mais ayant un léger décalage dû à son expertise (sévérité).

3.4.2.2 Algorithme Fuzzy C-means

Afin de réduire l'écart entre les évaluations des utilisateurs stricts et $\mathcal{M}_K$ nous utilisons l'algorithme Fuzzy C-means (1) proposé dans [1].

— **Entrées de l'algorithme**

L'algorithme prend en entrée : (i) un ensemble d'évaluations ($\mathcal{X}_{i=1,n}$) fournies par n utilisateurs ($\sqcap_{i=1,n}$) sur une ressource sur le Web R, (ii) le nombre de clusters flous à générer $Nb_{cluster}$ et enfin (iii) le critère de terminaison de l'algorithme ϵ représentant le critère de validité des clusters finaux. Un paramètre m contrôlant le degré de flou.

— **Sorties de l'algorithme**

L'algorithme produit en sortie : (i) un ensemble de clusters flous $\mathcal{C}_{j=1,Nb_{cluster}}$ et (ii) la matrice de degrés d'appartenance $\mathcal{ME}=\{\mathcal{ME}_{i,j}\}$ où $\mathcal{ME}_{i,j}$ représente le degré d'appartenance de $\mathcal{X}_{i=1,n}$ au cluster $\mathcal{C}_j$.

— **Fonction à minimiser**

L'algorithme vise à identifier les $\mathcal{ME}_{i,j}$ et les centroides $centroide(\mathcal{C}_j)$ minimisant la fonction :

$$\mathcal{J}_m = \sum_{i=1}^{n} \sum_{j=1}^{Nb_{cluster}} \mathcal{ME}_{ij}^m \times \left\| \mathcal{X}_i - centroid(\mathcal{C}_j) \right\|$$

— **Choix d'une mesure de similarité**

La distance euclidienne $\|*\|$ permet de mesurer la similarité entre deux évaluations.

Algorithme 1 : Fuzzy C-means

Entrée : $\mathcal{X}_{i=1,n}$, $Nb_{cluster}$, $m \in [0,1]$, ϵ

Sortie : $\mathcal{ME}$, $\mathcal{C}_{j=1,Nb_{cluster}}$

1 Initialiser $\mathcal{ME} \leftarrow \mathcal{ME}^{(0)}$ & $k \leftarrow 0$

2 Calculater $centroide(\mathcal{C}_j) = \sum_{i=1}^{n} (\mathcal{ME}_{ij}^m \times \mathcal{X}_i) / \sum_{i=1}^{n} \mathcal{ME}_{ij}^m$

3 Mettre à jour $\mathcal{ME}^{(k)}$, $\mathcal{ME}^{(k+1)}$

$$\mathcal{ME}_{ij} = 1 / \sum_{p=1}^{Nb_{cluster}} \left(\frac{\|\mathcal{X}_i - centroide(\mathcal{C}_j)\|}{\|\mathcal{X}_i - centroide(\mathcal{C}_p)\|^{\frac{2}{m-1}}} \right)$$

4 **si** $\left\| \mathcal{ME}^{(k)} - \mathcal{ME}^{(k+1)} \right\| < \epsilon$ **alors**
arrêt

sinon
$k = k + 1$, retour à l'étape 2

3.4.3 Illustration des techniques de clustering

Nous classifions maintenant l'ensemble des évaluations de l'exemple donné au début du chapitre (CF. sous-section 3.3.2) en utilisant respectivement le clustering classique et le clustering flou. Nous choisissons de représenter 3 clusters pour représenter trois classes d'avis sur la qualité de la ressource R : Mauvaise, Moyenne et Bonne.

3.4.3.1 Illustration l'algorithme K-Means

Le clustering des évaluations en utilisant l'algorithme $\mathcal{K}$-Means produit les trois clusters décris par la Table 3.2. Le cluster C_1 représente la classe d'avis des utilisateurs ayant fourni une mauvaise évaluation de la ressource R (0.2 sur une échelle de 0 à 1). Le cluster C_2 représente la classe d'avis des utilisateurs qui trouvent que la ressource est de qualité moyenne ou pas très bonne. La moyenne de ces évaluations est représentée par le centroide de C_2 : 0.71. Finalement le cluster C_3 représente la classe d'avis des utilisateurs qui trouvent que la ressource est de bonne qualité (0.86 sur une échelle de 0 à 1). Il s'agit du cluster le plus peuplé. Il est alors désigné comme le cluster majoritaire. Par conséquent l'avis de la majorité est 0.86 qui est le centroide de C_3.
L'algorithme $\mathcal{K}$-means détermine avec succès les fausses évaluations données par les utilisateurs malveillants u_1 et u_2 vu l'écart important existant entre l'avis de ces utilisateurs et l'avis de la majorité (0.86 - 0.2 = 0.66). Le défaut majeur de cet algorithme est qu'il exclut les avis des utilisateurs stricts u_3 et u_4 de l'avis de la majorité malgré leur faible écart de l'avis de la majorité (0.86 - 0.71 = 0.15).

3.4.3.2 Illustration de l'algorithme fuzzy C-means

Nous allons, maintenant, réaliser le clustering des mêmes évaluations par l'algorithme fuzzy C-means. Après plusieurs tests, nous

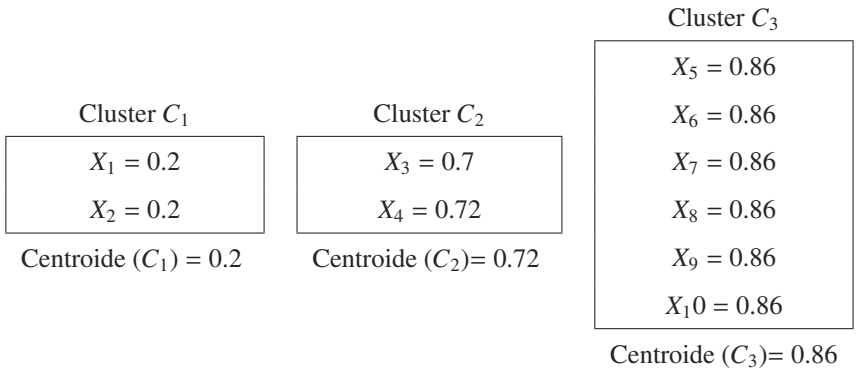


FIGURE 3.2 – Clustering des évaluations de la ressource R par $\mathcal{K}$-Means

avons fixé le critère de terminaison du clustering à 0.005, de sorte à obtenir des clusters bien définis mais en même temps qui se chevauchent. Comme les degrés d'appartenance initiaux sont attribués de manière aléatoire, l'algorithme s'arrête sur des valeurs aléatoires de degrés d'appartenance si le critère de terminaison est grand et donc avant que les clusters soient bien définis. Si le critère de terminaison est très petit le résultat se rapproche beaucoup de celui donné par l'algorithme $\mathcal{K}$-means.

Le clustering flou produit trois clusters flous. Les degrés d'appartenance de chaque évaluation à chacun de ces clusters sont représentés dans la Table 3.2. Par exemple, sur la troisième ligne, nous retrouvons les degrés d'appartenance de l'évaluation X_3 =0.2 aux différents clusters. X_3 n'appartient pas du tout au cluster C_1 (degré d'appartenance nul), appartient au cluster C_2 avec un fort degré d'appartenance égal à 0.98 et au cluster C_3 avec un faible degré d'appartenance égal à 0.02.

A première vue, nous remarquons que le centroide du cluster C_3 est passé de 0.86 à 0.84. Ceci est dû au fait que le calcul des centroides, dans le clustering flou, tient compte des évaluations n'appartenant pas complètement aux clusters. Ainsi les évaluations X_3et X_4 sont prises en compte lors du calcul du centroide de C_3, même si ces dernières y appartiennent avec des degrés d'appartenance faibles

(resp. 0.01 et 0.02).

		Clusters		
		C_1	C_2	C_3
centroide(C_j)		0.2	0.71	0.84
Évaluation X_i	$\mathcal{X}_1$=0.2	1.00	0.00	0.00
	$\mathcal{X}_2$=0.2	1.00	0.00	0.00
	$\mathcal{X}_3$=0.7	0.00	0.99	0.01
	$\mathcal{X}_4$ = 0.72	0.00	0.98	0.02
	$\mathcal{X}_5$ = 0.86	0.00	0.02	0.98
	$\mathcal{X}_6$ = 0.86	0.00	0.02	0.98
	$\mathcal{X}_7$ = 0.86	0.00	0.02	0.98
	$\mathcal{X}_8$ = 0.86	0.00	0.02	0.98
	$\mathcal{X}_9$ = 0.86	0.00	0.02	0.98
	$\mathcal{X}_{10}$ = 0.86	0.00	0.02	0.98

TABLEAU 3.2 – Matrice $\mathcal{ME}$ des degrés d'appartenance

3.4.4 Discussion

Le clustering des évaluations nous permet de détecter les classes d'avis parmi l'ensemble des évaluations fournies par les utilisateurs.

Dans le clustering classique, cette étape est suffisante pour retrouver le cluster majoritaire. En effet, il s'agit du cluster ayant le plus grand nombre d'éléments. Ce choix n'est plus valable avec le clustering flou. En effet, l'algorithme 1 génère un certain nombre de clusters ($Nb_{cluster}$) avec des frontières/limites floues. Les différents clusters se chevauchent et possèdent donc des éléments en commun. Ainsi le cluster ayant le plus grand nombre d'éléments ne représente pas toujours le cluster majoritaire et plus particulièrement si les degrés appartenance des éléments qu'il contient sont très faibles.

Il devient alors nécessaire d'identifier un nouveau *cluster majoritaire* (C_{Maj}) tenant en compte du fait que *toute évaluation possède un degré d'appartenance à un cluster*. Nous proposons dans ce qui suit stratégies pour le choix du cluster majoritaire C_{Maj}.

3.5 Recherche du cluster majoritaire

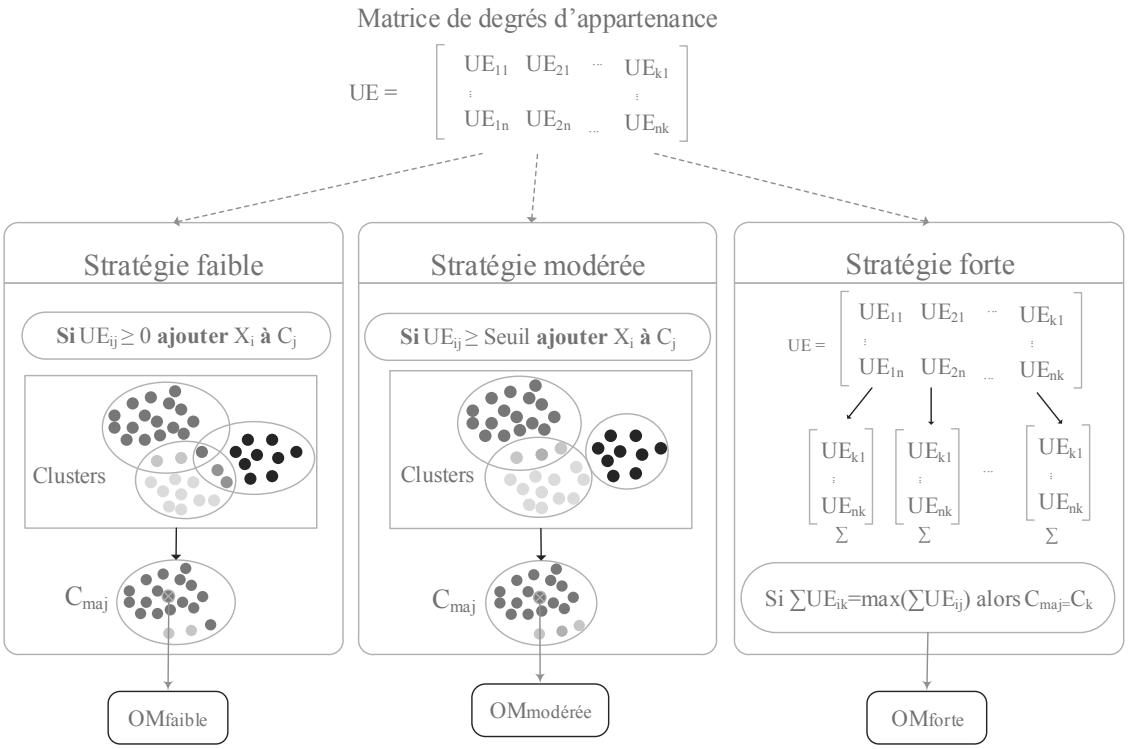


FIGURE 3.3 – Recherche du cluster majoritaire

Nous proposons trois stratégies pour la recherche du cluster majoritaire. Les stratégies reposent sur des valeurs qualitatives du degré d'appartenance à un cluster flou : stratégie faible, stratégie modérée, et stratégie forte. L'idée clé est de trouver le cluster le plus peuplé en raisonnant sur les degrés d'appartenance des évaluations. La figure 3.3 illustre les trois stratégies que nous proposons.

3.5.1 Stratégie faible

Dans la stratégie faible, une évaluation appartient à un cluster si son degré d'appartenance est strictement positif et n'y appartient pas si son degré d'appartenance à ce cluster est égal à 0. Elle conserve alors toutes les tailles actuelles des clusters. Le cluster le plus peuplé C_j est désigné comme le cluster majoritaire C_{Maj}. L'équation 3.1 identifie le *cluster majoritaire faible* (C_{Maj}^{faible}) comme :

$$C_{Maj}^{faible} = C_j, \left|C_j\right| = \max_{\forall k=1,Nb_{cluster}} (|C_k|) \wedge \mathcal{ME}_{i,k} > 0, i \in [1,n] \tag{3.1}$$

La figure 3.4 (a) montre le résultat d'application de cette stratégie sur l'ensemble des évaluations de l'exemple fourni dans la Table 3.1. La stratégie faible identifie trois clusters. Deux parmi ces clusters sont de taille maximale incluant les évaluations de X_3 à X_{10}. Le choix du cluster majoritaire est alors remis au hasard. Comme les clusters contiennent le même ensemble d'évaluations, la différence serait alors au niveau de l'avis de la majorité qui sera égale à 0.71 si C_2 est choisi et 0.84 si C_3 est choisi. La première représente de manière appropriée les évaluations des utilisateurs stricts et la deuxième représente l'avis de la majorité réelle avec un décalage vers l'avis des utilisateurs stricts.

L'inconvénient de cette stratégie est que plus les degrés d'appartenance aux clusters sont faibles, plus le recouvrement des clusters est important. Dans certains cas, tous les clusters peuvent contenir toutes les évaluations et sont par conséquent tous de taille maximale. Lorsque le critère de terminaison du clustering ϵ n'est pas assez faible, les clusters flous représenteront un recouvrement total où tous les degrés d'appartenance sont non nuls. Il devient alors difficile de déterminer le cluster majoritaire. Pour remédier à ce problème, nous proposons la stratégie modérée.

3.5.2 Stratégie modérée

La stratégie modérée garde une évaluation dans un cluster seulement si son degré d'appartenance est supérieur à un seuil γ et sélectionne ensuite le cluster le plus peuplé C_j pour devenir C_{Maj}. L'équation 3.2 identifie le *cluster majoritaire modéré* ($C_{Maj}^{modérée}$) :

$$C_{Maj}^{modérée} = C_j, \ |C_j| = \max_{\forall k=1,Nb_{cluster}} (|C_k|) \\ \wedge \mathcal{ME}_{i,k} \geq \gamma, i \in [1,n] \tag{3.2}$$

Dans la figure 3.4(b), la stratégie modérée arrive à retrouver le cluster majoritaire modéré $C_{Maj}^{modérée}$ composée des évaluations de X_5 à X_6. Ce cluster correspond parfaitement au cluster majoritaire réel, tout en prenant en compte l'avis des utilisateurs stricts u_3 et u_4 avec le décalage léger du centroide de $C_{Maj}^{modérée}$ de 0.86 à 0.84.

3.5.3 Stratégie forte

Les deux stratégies précédentes dépendent fortement de la répartition des évaluations dans les différents clusters obtenus par le clustering flou. Le cluster majoritaire est facilement identifiable en examinant visuellement les résultats.
Cependant cette répartition peut porter à confusion lors d'un calcul automatique. Nous avons cherché à trouver une stratégie plus robuste basée uniquement sur les degrés d'appartenances des évaluations afin d'être sûr de sélectionner le cluster le plus peuplé.
La stratégie forte identifie le cluster majoritaire C_{Maj} comme étant le cluster ayant le plus haut degré d'appartenance de l'ensemble des évaluations. L'équation 3.3 identifie le *cluster majoritaire fort* :

$$C_{Maj}^{forte} = C_j, \ \sum_{i\in[1,n]} (\mathcal{ME}_{i,j}) = \max_{\forall k=1,Nb_{cluster}} (\sum_{i\in[1,n]} (\mathcal{ME}_{i,k})) \tag{3.3}$$

De même, cette stratégie a réussi à retrouver le cluster majoritaire réel. La figure 3.4(c) montre la somme des degrés d'appartenance de chaque cluster. Le cluster C_3 possède la plus haute somme des

degrés d'appartenance de ses éléments : 5.91. Il s'agit alors du cluster majoritaire fort.

Cluster C_1	Cluster C_2	Cluster C_3
$X_1 = 0.2$	$X_3 = 0.7$	$X_3 = 0.7$
$X_2 = 0.2$	$X_4 = 0.72$	$X_4 = 0.72$
	$X_5 = 0.86$	$X_5 = 0.86$
	$X_6 = 0.86$	$X_6 = 0.86$
	$X_7 = 0.86$	$X_7 = 0.86$
	$X_8 = 0.86$	$X_8 = 0.86$
	$X_9 = 0.86$	$X_9 = 0.86$
	$X_10 = 0.86$	$X_{10} = 0.86$
Centroide (C_1) = 0.2	Centroide (C_2)= 0.71	Centroide (C_3)= 0.84

(a) Cluster majoritaire faible $C_{Maj}^{faible} = C_2$ ou C_3

Cluster C_1	Cluster C_2	Cluster C_3
$X_1 = 0.2$	$X_3 = 0.7$	$X_3 = 0.7$
$X_2 = 0.2$	$X_4 = 0.72$	$X_4 = 0.72$
	$X_5 = 0.86$	$X_5 = 0.86$
	$X_6 = 0.86$	$X_6 = 0.86$
	$X_7 = 0.86$	$X_7 = 0.86$
	$X_8 = 0.86$	$X_8 = 0.86$
	$X_9 = 0.86$	$X_9 = 0.86$
	$X_{10} = 0.86$	$X_{10} = 0.86$
Centroide (C_1) = 0.2	centroide (C_2)= 0.71	Centroide (C_3)= 0.84

(b) Cluster majoritaire modéré : $C_{Maj}^{modérée} = C_3$

	C_1	C_2	C_3
$\sum_{i\in[1,n]} (\mathcal{ME}_{i,j})$	2	2.09	5.91

(c) Cluster majoritaire fort : $C_{Maj}^{forte} = C_3$

FIGURE 3.4 – Recherche du cluster majoritaire par les différentes stratégies

3.6 Calcul de la crédibilité

Le cluster majoritaire étant établi, la prochaine étape est de calculer la crédibilité des utilisateurs par rapport au cluster majoritaire. La crédibilité de l'utilisateur dans notre modèle se définit par la similarité entre son avis et l'avis de la majorité. Les mesures de similarité les plus utilisées sont les mesures de distance. Dans notre cas, nous avons besoin d'une mesure de distance pour des champs continus. Pour ce type de données, la distance euclidienne reste la plus utilisée (Équation 3.4). Elle est un cas particulier (p=2) de la métrique de Minkowski. Ces mesures posent un problème lorsque les échelles des données ne sont pas homogènes. Le mieux est alors d'utiliser la forme normalisée de cette distance.

$$d_2(x_i, x_j) = (\sum_{k=1}^{d}(x_{i,k} - x_{j,k})^2)^{1/2} = \left\| x_i - x_j \right\|_2 \quad avec x_i = (x_{i,1}, ..., x_{i,d}) \ \ et \ \ x_j = (x_{j,1}, ..., x_{j,d}) \tag{3.4}$$

L'équation 3.5 définit la crédibilité de utilisateur u_i comme étant la distance entre son évaluation et l'avis de la majorité représentée par le centroide de C_{Maj}. Cette crédibilité est calculée en utilisant la distance euclidienne normalisée $\|*\|_{\mathcal{N}}$ comme mesure de similarité :

$$\mathcal{CR}_i^j = 1 - \left\| \mathcal{X}_i - centroide(C_{Maj}^{stratégie}) \right\|_{\mathcal{N}}, \quad stratégie \in \{faible, modérée, forte\} \tag{3.5}$$

La Table 3.3 montre Résultats de l'estimation de la crédibilité des utilisateurs u_1 .. u_{10} en utilisant le clustering classique en comparaison avec le clustering flou selon les différentes stratégies. Une crédibilité égale à 1 signifie que l'utilisateur est complètement crédible ; égale à 0 signifie qu'il ne l'est pas du tout et plus sa crédibilité s'approche de 0 moins il est crédible. Selon la stratégie faible, les utilisateurs stricts sont les plus crédibles. Selon le clustering

utilisateur u_i	évaluation X_i	crédibilité $\mathcal{CR}_i^j$			
		$\mathcal{K}$-means	stratégie faible	stratégie modérée	stratégie forte
u_1	$\mathcal{X}_1$=0.2	0.34	0.49	0.36	0.36
u_2	$\mathcal{X}_2$=0.2	0.34	0.49	0.36	0.36
u_3	$\mathcal{X}_3$=0.7	0.84	0.99	0.86	0.86
u_4	$\mathcal{X}_4$=0.71	0.86	0.99	0.88	0.88
u_5	$\mathcal{X}_5$=0.86	1	0.85	1	1
u_6	$\mathcal{X}_6$=0.86	1	0.85	1	1
u_7	$\mathcal{X}_7$=0.86	1	0.85	1	1
u_8	$\mathcal{X}_8$=0.86	1	0.85	1	1
u_9	$\mathcal{X}_9$=0.86	1	0.85	1	1
u_{10}	$\mathcal{X}_1$0=0.86	1	0.85	1	1

TABLEAU 3.3 – Résultats de l'estimation de la crédibilité des utilisateurs

classique et les stratégies modérée et forte les utilisateurs X_5 à X_{10} sont les plus crédibles.
Nous observons qu'indépendamment de la stratégie choisie la crédibilité des utilisateurs X_3 et X_4 strictes a augmenté. Nous remarquons aussi que lorsqu'on considère l'avis des utilisateurs strictes comme étant le plus fiable, tel le cas de la stratégie faible, nous prenons le risque d'augmenter la crédibilité des utilisateurs malveillants X_1 et X_2 qui passe d'environ 0.34 à 0.49. Il est alors plus judicieux d'utiliser la stratégie modérée ou forte.

3.7 Approche déterministe de la confiance basée sur la crédibilité

Soit $\mathcal{L}_i$ l'ensemble des utilisateurs u_k émettant des évaluations $\mathcal{X}_k^j$ sur les performances d'une ressource R_j. u_i estime la confiance $C_{\mathcal{X}_{\mathcal{L}_i}^j}$ d'une ressource R_j en fonction des évaluations $\mathcal{X}_k^j$ et des crédibilités $\mathcal{CR}_k$. L'équation 3.6 calcule la confiance $C_{\mathcal{X}_{\mathcal{L}_i}^j}$ comme une

moyenne pondérée de $\mathcal{X}_k^j$.

$$C_{\mathcal{X}_{\mathcal{L}_i}^j} = \frac{1}{\sum\limits_{k \in \mathcal{L}_i} \mathcal{CR}_k} \times \sum_{k \in \mathcal{L}_i} (\mathcal{CR}_k \times \mathcal{X}_k^j) \tag{3.6}$$

$C_{\mathcal{X}_{\mathcal{L}_i}^j}$ est une "bonne" mesure de confiance, mais ne permet pas d'affirmer que R_j est la plus fiable en raison de la connaissance limitée de $\mathcal{L}_i$ sur R_j. Une solution serait que u_i consulte un ensemble supplémentaire de pairs ($\mathcal{L'}_i$) ayant au préalable établi des valeurs de confiance de R_j à partir d'autres évaluations données par $u_{m \neq k}$. Par conséquent, la confiance peut être également calculée en utilisant la réputation du R_j ($\mathcal{REP}_j^i$). Dans [34], Sabater et Sierra définissent la réputation comme étant l'avis d'une personne sur un objet. Ramchurn et al. affinent cette définition en précisant que "...*Un avis peut être principalement établi à partir de l'agrégation des avis des membres de la communauté sur l'un d'entre eux*" [32]. Ramchurn et al. différencient entre la confiance et la réputation ; la première est issue des interactions directes et la seconde est obtenue de l'environnement ou des autres agents permettant d'établir la confiance. L'équation 3.7 calcule la réputation $\mathcal{REP}_j^i$:

$$\mathcal{REP}_j^i = \frac{1}{|\mathcal{L'}_i|} \times \sum_{m \in \mathcal{L'}_i} C_{\mathcal{L}_m}^j \tag{3.7}$$

Finalement, l'équation 3.8 combine à la fois $C_{\mathcal{X}_k^j}^i$ et $\mathcal{REP}_j^i$ pondérées respectivement par les scores de préférences de u_i, α et β.

$$C_j^i = \alpha \times C_{\mathcal{X}_{\mathcal{L}_i}^j} + \beta \times \mathcal{REP}_j^i \tag{3.8}$$

3.8 Conclusion

Dans ce chapitre, nous avons proposé un modèle de crédibilité résiliant aux attaques des évaluations biaisées. Afin de limiter l'impact de ces évaluations sur le calcul de la confiance des ressources sur le

Web, ce modèle estime la crédibilité des utilisateurs les ayant fournies. Le modèle se base principalement sur deux notions : l'avis de la majorité et les utilisateurs stricts. Notre principal objectif était de réduire l'écart entre l'avis de la majorité et les utilisateurs stricts souvent exclus par les systèmes existants d'évaluation de la confiance. Nous avons réussi à inclure les évaluations de ces derniers dans l'avis de la majorité en utilisant la technique clustering flou.

Nous également proposé une approche déterministe basée sur la crédibilité pour l'évaluation de la confiance des ressources sur le Web. Elle consolide les différentes évaluations collectées en prenant en compte la crédibilité des évaluateurs. Deux mesures ont été proposées : les évaluations des utilisateurs et la réputation des ressources sur le Web.

Dans le chapitre suivant nous nous sommes fixés un nouveau challenge qui est la résolution du problème de l'attaque Sybil.

Chapitre 4

Modèle de filtrage des évaluations tolérant aux attaques Sybils

Sommaire

4.1 Introduction

Dans cette section, nous proposons un modèle filtrage des évaluations tolérant aux attaques Sybil. Il s'agit d'une attaque réalisée par un seul utilisateur réel mais possédant plusieurs identités virtuelles dans un réseau social. Cet utilisateur malveillant est capable d'influencer le résultat de la confiance d'une ressource en fournissant les mêmes évaluations à travers ses identités virtuelles. Pour détecter les utilisateurs Sybil, une solution, serait d'analyser leurs profils des utilisateurs (i.e., les différentes informations personnelles qu'ils fournissent lors de leur inscription tel que le nom, le prénom, l'adresse mail, etc).

Cependant, cette solution ne respecte pas la vie privée des utilisateurs. Nous avons alors exploré les approches existantes persévérant cette vie privée. De telles approches se basent sur la topologie du réseau social auxquels sont inscrits les utilisateurs et plus précisément des liens sociaux entre ces derniers. Un lien social représente une relation de confiance entre deux utilisateurs. Il est alors difficile pour un utilisateur Sybil de créer plusieurs liens avec des utilisateurs réels. Les approches existantes utilisent cette règle pour filtrer des utilisateurs Sybil et réduire leurs capacités d'attaque.

Néanmoins, ces approches se basent uniquement sur les relations de confiance entre les utilisateurs et sur des choix arbitraires dans leur méthode de filtrage. Elles ne s'intéressent pas aux caractéristiques des utilisateurs et plus particulièrement leur crédibilité. En effet, un utilisateur peut être non Sybil sans pour autant être crédible dans son évaluation. Ce dernier peut toujours émettre une fausse évaluation à partir de son vrai compte utilisateur sans être détecté par l'approche de filtrage.

Notre modèle combine les deux notions de réseaux sociaux et de la crédibilité afin de filtrer les évaluations. Le modèle de filtrage utilise une première technique, celle de la distribution des capacités d'évaluations. Cette technique distribue des capacités d'éva-

luations aux utilisateurs de sorte à former un graphe de confiance. Elle a pour but d'augmenter la capacité des utilisateurs crédibles à émettre leurs évaluations et limiter la capacité d'attaque des utilisateurs Sybil. Ensuite le modèle de filtrage utilise une deuxième technique, celle de recherche dans les graphes afin de sélectionner un nombre maximal d'utilisateurs crédibles et non Sybil. Les deux techniques utilisent la crédibilité comme heuristique.
Le présent chapitre est organisé comme suit. Une première section décrit la formalisation du réseau social. Une deuxième section décrit la conception de notre modèle et plus particulièrement les différentes phases de sélection des utilisateurs crédibles et non Sybil.

4.2 Fondements de notre modèle de filtrage des évaluations

4.2.1 Topologie du réseau social

Dans notre modèle, les utilisateurs s'inscrivent dans un réseau social avec une identité pour avoir le droit de fournir des évaluations des ressources sur le Web avec lesquels ils interagissent. Un utilisateur u_i peut également établir des liens de confiance avec d'autres utilisateurs (u_k). Un lien de confiance représente la croyance de u_i que u_k ne soit pas un utilisateur Sybil et qu'il évaluera honnêtement la ressource.
L'étude de la topologie du réseau social révèle la présence de régions d'utilisateurs honnêtes connectées à d'autres régions d'utilisateurs Sybil à travers un nombre limité de liens d'attaques (lien reliant un noeud honnête à un noeud Sybil) comme le montre la figure 4.1. Ceci est dû au fait que cela nécessite beaucoup d'efforts humains pour qu'un utilisateur puisse mettre en confiance un autre utilisateur et créer un lien social avec ce dernier. Par ce fait, il est difficile pour un attaquant d'établir un très grand nombre de

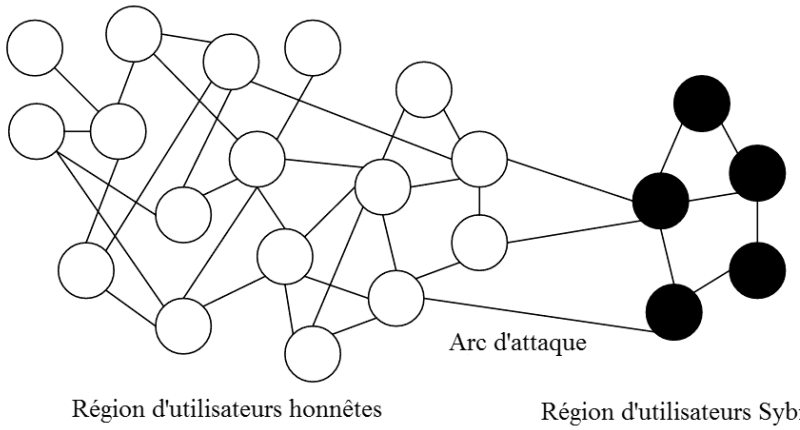


FIGURE 4.1 – Topologie d'un réseau social en présence d'utilisateurs Sybil

liens de confiance avec les utilisateurs honnêtes, cependant, ce dernier peut créer de multiples liens avec d'autres utilisateurs Sybil. Le but de notre modèle est alors de cibler les régions d'utilisateurs honnêtes.

4.2.2 Vue globale de notre approche de filtrage

Nous proposons une approche de filtrage des évaluations basée sur la topologie du réseau social et sur la crédibilité de l'utilisateur (CF. chapitre 3). Le filtrage est réalisé en ciblant une région d'utilisateurs honnêtes dans le réseau social. Dans [4] Cheng et Friedman démontrent qu'une approche d'évaluation de la confiance utilisant seulement la topologie du réseau social ne permet pas de distinguer les utilisateurs ayant une seule identité de ceux ayant de multiples identités. Il est donc nécessaire de connaître au moins un noeud digne de confiance communément appelé `source`. L'exploration du voisinage du noeud source permet de localiser une des régions des utilisateurs honnêtes.
La Figure 4.2 illustre les quatre phases de notre modèle. La première étape est l'élagage du réseau social. Cette dernière a pour but de diminuer le nombre d'arcs d'attaques sans pour autant atteindre

les utilisateurs honnêtes et la chance/potentiel que leur évaluation sera prise en compte. La deuxième étape consiste à construire à partir du réseau élagué, un graphe de confiance étiqueté par des valeurs positives. Les valeurs sont attribuées aux arcs de sorte à augmenter le pouvoir d'évaluer des utilisateurs honnêtes et diminuer celui des utilisateurs malhonnêtes tout en respectant une limite C_{max}. Le graphe est maintenant prêt à être exploré dans la troisième étape qui a pour but de sélectionner les utilisateurs dont les évaluations seront prises en compte. Ceci est réalisé au moyen d'un algorithme de calcul du flot maximal.

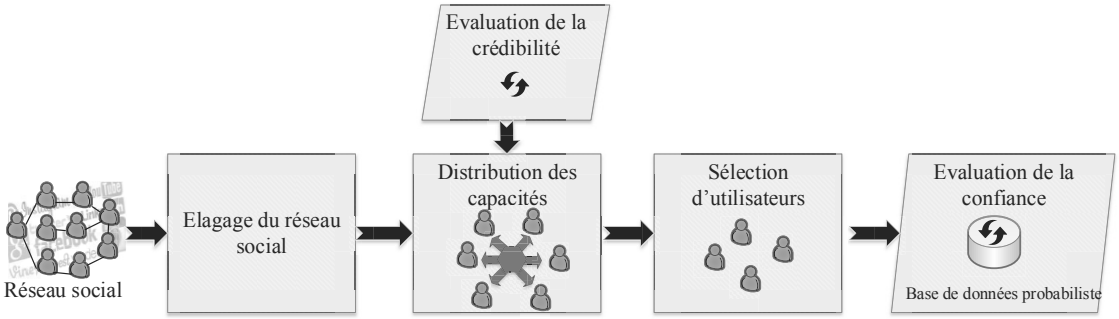


FIGURE 4.2 – Aperçu de l'approche

4.2.3 Formalisation du graphe de confiance

Formellement, le graphe de confiance G est un tuple $< S, N, A, Cr^N, C^A >$ où :

1. S représente la `source`,
2. $N = \{N_1, N_2, \dots\}$ est un ensemble de noeuds qui correspondent aux identités des utilisateurs,
3. $A = \{A^{(N_1,N_2)}, \dots, A^{(N_i,N_j)}, \dots\}$ est un ensemble d'arcs (i.e, liens de confiance) entre les noeuds,
4. $Cr^N : N \rightarrow [0, 1]$ affecte une valeur de crédibilité à chaque N_i,
5. $C^A : A \rightarrow \mathbb{N}$ affecte une valeur de capacité à chaque $A^{(N_i,N_j)}$.

4.3 Élagage du réseau social

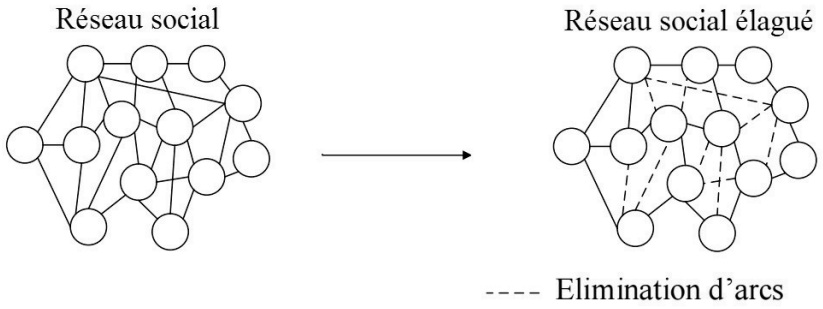


FIGURE 4.3 – Élagage du réseau social

Dans notre modèle, notre but est non de détecter les utilisateurs Sybil mais de les éviter. En effet, se baser sur la topologie du réseau social, ne permet pas de déterminer avec certitude si un noeud correspond à un utilisateur honnête ou Sybil vu que nous ne disposons d'aucune information sur son identité. L'idée est alors de trouver un compromis entre limiter la capacité d'attaque d'un noeud dans le cas où le noeud est Sybil tout en laissant au noeud honnête sa chance d'exprimer son opinion.
Dans notre approche la capacité d'évaluation d'un utilisateur Sybil est égale à la capacité globale des arcs entrants à cet utilisateur. Cette capacité d'évaluation correspond aussi à sa capacité à fournir des fausses évaluations et correspond alors à sa capacité d'attaque. Par conséquent si nous diminuons d'arcs entrants au niveau d'un noeud Sybil, le taux des fausses évaluations que la `source` sera susceptible de collecter diminue. Nous adoptons la solution proposée dans [42] qui suggère de limiter le nombre d'arcs entrants par noeud à un seuil prédéfini $e_{entrant}$. Ainsi le nombre d'arcs d'attaques d'un utilisateur Sybil est limité à $e_{entrant}$. Le choix de $e_{entrant}$ serait motivé lors de la mise en oeuvre de notre approche (CF. Chapitre 6).
L'élagage présente deux avantages : (i) éliminer des chemins redondants dans le graphe de confiance et ainsi accélérer la collecte

des évaluations et (ii) éviter le cas où un utilisateur Sybil possède des multiples liens avec des utilisateurs honnêtes et peut par conséquent émettre des fausses évaluations à travers l'un de ces liens.
De plus, cette technique peu susceptible d'affecter la capacité des utilisateurs honnêtes à évaluer. En effet, nous distribuons les capacités d'évaluation de telle sorte que chaque noeud honnête a la possibilité d'émettre une seule évaluation à travers un chemin de la `source`, et ce à travers de ses $e_{entrant}$ arcs entrants.

Algorithme 2 : Elagage du réseau social

Entrée : $G = < S, N, A, Cr^N > e_{entrant}$
Sortie : G élagué

```
file ← CréerFile()
file.enfiler(S)
marquer(S)
Tant que ( !file.vide() ) faire
    courant ← file.défiler()
    k = card (parents(courant))
    Si k > e_entrant alors
        x = k - e_entrant
        supprimerArcsAléatoirement/supprimerArcsSelonCrédibilité (x, courant)
    Fin Si
    supprimerArcsEntreFils(courant)
    Pour tout N_i ∈ fils(courant) et nonMarqué(N_i) faire
        marquer(N_i)
        enfiler(N_i)
    Fin Pour
FinTant que
return G
```

Pour réaliser l'élagage du réseau social nous utilisons l'algorithme 2. L'algorithme considère le réseau social comme un graphe et effectue un parcours en largeur ce dernier. Au niveau de chaque noeud parcouru, l'algorithme supprime les arcs entrants en plus afin de

limiter leur nombre à $e_{entrant}$.

Le parcours se fait en utilisant une file. La première étape consiste à mettre le noeud source dans la file (le défiler). Ensuite retirer le noeud se trouvant au début de la file pour l'élaguer. L'élagage se fait en deux étapes. La première étape consiste à supprimer les arcs entre fils de sorte à éliminer tous les liens existants entre les noeuds du même niveau en utilisant l'algorithme 3. Ce traitement permet d'éviter de parcourir deux fois le même noeud et d'éliminer par conséquent les chemins redondants. La deuxième étape consiste à supprimer les arcs en plus afin de limiter le nombre d'arcs entrants à un noeud courant à $e_{entrant}$. La suppression se fait ou bien aléatoirement en utilisant l'algorithme 5 ou bien en utilisant la crédibilité comme heuristique en utilisant l'algorithme 4. Les arcs supprimés sont ceux reliant le noeud courant aux noeuds ayant la plus faible valeur de crédibilité.

L'étape d'après consiste à mettre tous les voisins non explorés (les fils du noeud courant) à la fin de la file (les enfiler). Ces étapes sont répétées tant que la file est non vide. Les noeuds qui sont déjà visités sont marqués afin d'éviter qu'un noeud soit exploré plusieurs

fois.

Algorithme 3 : supprimerArcsEntreFils

Entrée : Entier x, Noeud courant
Sortie :
  Pour $N_i \in$ fils(courant) **faire**
    Pour $N_j \in$ fils(courant) **faire**
      Si $\exists A^{(N_i,N_j)}$ **alors**
        supprimer ($A^{(N_i,N_j)}$)
      Fin Si
    Fin Pour
  Fin Pour

Algorithme 4 : supprimerArcsSelonCrédibilité

Entrée : Entier x, Noeud courant
Sortie :
  P ← parents(Courant)
  trierParOrdreCroissantSelonCrédibilité(P)
  N ← 0
  Tant que N < x) **faire**
    SupprimerPremierElement(P)
    N ← N + 1
  FinTant que

Algorithme 5 : supprimerArcsAléatoirement

Entrée : Entier x, Noeud courant
Sortie :
  P ← parents(courant)
  Pour i de 1 à x **faire**
    P_i ← random(P)
    supprimer ($A^{(P_i,courant)}$)
    supprimer (P_i, P) ;
  Fin Pour

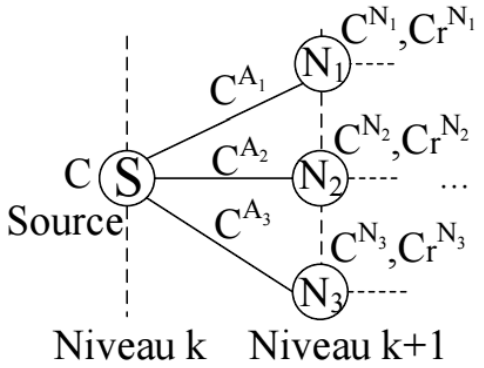


FIGURE 4.4 – Construction du graphe de confiance

4.4 Construction du graphe de confiance

Après avoir diminué la capacité d'attaque des utilisateurs Sybil en élaguant le réseau social (Algorithme 2), cette étape vise à augmenter la capacité d'évaluation des utilisateurs crédibles ayant une seule identité. Afin d'initialiser la capacité d'évaluation des utilisateurs crédibles et ayant une seule identité, nous développons un nouveau mécanisme de distribution de tickets à partir de la *source* basé sur la crédibilité des utilisateurs (CF. Chapitre 3). Un ticket correspond à une capacité d'évaluation. La distribution des tickets est faite en réalisant un parcours en largeur du réseau social de sorte à former un graphe de confiance où chaque utilisateur possède un ou plusieurs tickets. A l'issue de cette étape seuls les utilisateurs possédant des tickets ont le droit d'émettre leur évaluations.

Nous définissions les niveaux des noeuds par rapport à leur distance à la source, un noeud se trouvant à un niveau k possède alors au moins un parent au niveau $k-1$. Le noeud N_i reçoit C tickets de la part des arcs le reliant aux noeuds parents ($Parent_i$) se trouvant aux niveaux inférieurs, il consomme un ticket et redistribue les tickets restants aux noeuds fils ($Fils_i$) aux niveaux supérieurs.

Nous proposons une stratégie de distribution de tickets où un noeud N_k redistribue C^{N_k} tickets à travers ses arcs sortants selon l'équation 4.1. Afin de conserver le flot dans le graphe de confiance, nous

arrondissons les valeurs de capacité à la valeur entière inférieure ou supérieure.

$$C^{A^{(N_k,N_i)}} = arrondi(\frac{C^{N_k} * Cr_i}{\sum_{\forall N_j \in Fils_k} Cr_j}) \tag{4.1}$$

Nous associons à N_i une capacité C^{N_i} en utilisant l'équation 4.2. C^{N_i} est égale à la capacité des arcs entrants à N_i.

$$C^{N_i} = \sum_{\forall N_k \in Parent_i} C^{A^{(N_k,N_i)}} - 1 \tag{4.2}$$

Algorithme 6 : Distribution des capacités

Entrée : $G = < S, N, A, Cr^N, C^A > e_{entrant}$ et C_{max}

Sortie : G étiqueté par les capacités

```
file ← CréerFile()
file.enfiler(S)
marquer(S)
C^S ← C_max
Tant que ( !f.vide () ) faire
  courant ← file.défiler()
  Pour N_i ∈ fils(courant) et nonMarqué(N_i) faire
    Équation 4.1
    Si ( C^A(courant,N_i) = 0) alors
      supprimer (C^A(courant,N_i))
    Sinon
      Équation 4.2
      marquer(N_i)
      enfiler(N_i)
    Fin Si
  Fin Pour
FinTant que
return G
```

4.5 Sélection d'utilisateurs

Cette étape vise à sélectionner le nombre maximal d'utilisateurs crédibles et ayant une seule identité dans le graphe de confiance. À cet effet, nous adaptons l'algorithme de flot maximal proposé par Ford et Fulkerson [9] en utilisant la recherche heuristique pour guider la sélection vers ces utilisateurs.

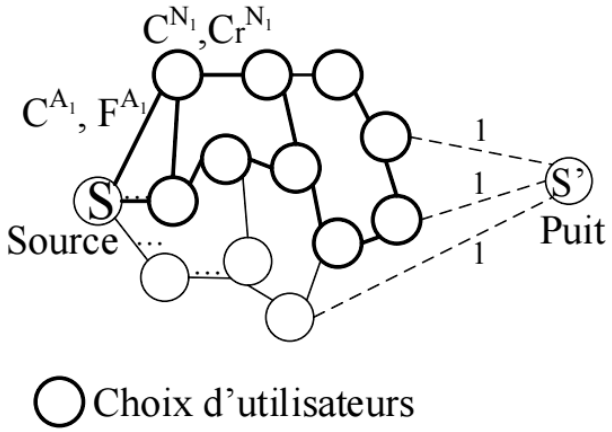


FIGURE 4.5 – Sélection d'utilisateurs dans le graphe de confiance

Après la construction du graphe de confiance, l'étape suivante est de sélectionner un nombre élevé d'utilisateurs crédibles ayant une seule identité. Nous formalisons cette condition de sélection comme problème de flot maximal. Ce problème revient à maximiser le flot passant par les chemins allant d'une `source` S à un `puit` S' dans le graphe de confiance. Afin d'appliquer l'algorithme de flot maximal un `puit` est ajouté au graphe de confiance en le liant aux feuilles. Les arcs entre le `puit` et les feuilles sont étiquetés par 1. Nous nous sommes inspirés de l'algorithme de Ford-Fulkerson en définissant trois fonctions :

(i) $F : A \rightarrow \mathbb{N}$ affecte à chaque arc $A^{(N_i,N_j)}$ ayant une capacité $C^{A^{(N_i,N_j)}}$ une valeur de flot $F(A^{(N_i,N_j)}) \leq C^{A^{(N_i,N_j)}}$. $F(A^{(N_i,N_j)})$ représente le flot qui pourrait être passé par $A^{(N_i,N_j)}$.

(ii) $R_F : A \rightarrow \mathbb{N}$ traite la capacité résiduelle. R_F représente le

flot restant qui pourrait être passé par $A^{(N_i,N_j)}$ ($R_F(A^{(N_i,N_j)}) = C^{A^{(N_i,N_j)}} - F^{A^{(N_i,N_j)}}$).

(iii) G_F établit le graphe résiduel qui représente l'état du graphe quand un certain flot est passé par un de ses arcs. Formellement, G_F est représentée par un tuple $< S, N, A_F, Cr^N, C^A >$, où $A_F = \{A^{(N_i,N_j)} \in A \backslash R_F(A^{(N_i,N_j)}) \geq 0\}$.

Algorithme 7 : Algorithme de sélection d'utilisateurs

Entrée : $G = < S, N, A, Cr^N, C^A >$ et S'

Sortie : F, U

Pour chaque $A^{(N_i,N_j)} \in A$ **faire**

$F(A^{(N_i,N_j)}) \leftarrow 0$

Fin Pour

$U \leftarrow \emptyset$

Répéter

chemin = trouverCheminMeilleurEnPremier(G_F, S')

Pour chaque $A^{(N_i,N_j)} \in$ *chemin* **faire**

$F(A^{(N_i,N_j)}) \leftarrow F(A^{(N_i,N_j)}) + 1$

$F \leftarrow F + 1$

$U \leftarrow U \cup \{N_i, N_j\}$

Fin Pour

Jusqu'à $chemin = NULL$

return F, U

Lorsque $R_F(A^{(N_i,N_j)})$ est égale à 0, $A^{(N_i,N_j)}$ est considéré comme saturé et ne pouvant pas faire passer plus de flot. Pour trouver un chemin augmentant le flot, il existe des méthodes de recherche de chemin telles que la recherche en profondeur utilisée dans [42].
Cependant, ces méthodes de recherche se basent uniquement sur la structure du graphe. Nous proposons ainsi l'Algorithme 7 qui utilise la recherche du meilleur en premier (*trouverCheminMeilleurEnPremier*).
L'algorithme du recherche du meilleur en premier reposant sur le principe de l'algorithme de recherche A* proposé dans [30]. La stratégie consiste à partir du noeud source, ordonner les noeuds

fils selon un critère de d'évaluation, choisir le meilleur noeud et le mémoriser, continuer ce processus jusqu'à arriver au noeud destination. Le critère d'évaluation dans notre modèle est la crédibilité de l'utilisateur (voir Algorithme 8).

L'Algorithme 7 retourne le flot maximal F et l'ensemble des utilisateurs U sélectionnés pour l'évaluation. L'objectif est d'atteindre un optimum global en se basant sur un choix optimal local (i.e, un chemin de la `source` au `puit` contenant le nombre maximal d'utilisateurs crédibles et ayant une seule identité). À chaque itération l'Algorithme 7 commence à partir de la `source` et augmente le chemin de flot au prochain niveau en choisissant le noeud le plus crédible N_i ayant la valeur la plus basse de la fonction heuristique $h(N_i) = 1/Cr^{N_i}$ afin de la minimiser et une capacité résiduelle non nulle ($R_F(E^{(S,N_i)}) > 0$).

Algorithme 8 : Algorithme de recherche du meilleur chemin

Entrée : $G = < S, N, A, Cr^N, C^A >$ et S'

Sortie : *meilleurChemin*

```
Fin ← Faux
chemin ∪ {S}
Tant que ( ! Fin ) faire
    meilleur ← MeilleurHeuristique1/MeilleurHeuristique2(courant)
    courant ← meilleur
    chemin ∪ {courant}
    Si courant = S' alors
        Fin ← Vrai
    Fin Si
FinTant que
return chemin
```

Algorithme 9 : TrouverMeilleurHeuristique1

Entrée : Noeud courant
Sortie : Noeud meilleur
N ← fils(courant)
meilleur ← premierElement(N)
Pour $N_i \in$ N **faire**
Si ($Cr^{N_i} > Cr^{meilleur}$) et ($R_F(E^{(courant,N_i)}) > 0$ **alors**
meilleur ← N_i
Fin Si
Fin Pour
Retourner meilleur

Algorithme 10 : TrouverMeilleurHeuristique2

Entrée : Noeud courant
Sortie : Noeud meilleur
N ← fils(courant)
meilleur ← premierElement(N)
Pour $N_i \in$ N **faire**
Si $Cr^{N_i} > Cr^{meilleur}$ et $R_F(E^{(courant,N_i)}) > 0$ **alors**
meilleur ← N_i
SinonSi $Cr^{N_i} = Cr^{meilleur}$ et $R_F(E^{(courant,N_i)}) > 0$ et $C^{N_i} > C^{meilleur}$ **alors**
meilleur ← N_i
Fin Si
Fin Pour
Retourner meilleur

4.6 Conclusion

Dans ce chapitre, nous avons proposé une approche pour lutter contre les attaques Sybil lors du calcul de la confiance des ressources sur le Web. Pour démystifier les utilisateurs Sybil, l'approche se base sur les utilisateurs crédibles fournissant des évaluations cohérentes et par conséquent, dignes de confiance. Les

utilisateurs Sybil et ceux ayant une seule identité font partie d'un graphe auquel nous appliquons un mécanisme de distribution de tickets pour évaluer les ressources sur le Web en termes de propriétés non-fonctionnelles et un autre de sélection d'évaluations pertinentes basé sur une adaptation de l'algorithme de flot maximal. Les deux mécanismes sont basés sur la crédibilité de l'utilisateur. L'objectif est de sélectionner un nombre élevé d'utilisateurs crédibles ayant une seule identité et de diminuer la capacité d'attaque des utilisateurs Sybil.

Chapitre 5

Modèle probabiliste de confiance

Sommaire

5.1 Introduction

Dans ce chapitre, nous présentons le modèle d'évaluation de la confiance d'une ressource sur le Web qui constitue l'objectif final de notre travail. En fait, il s'agit de trouver une solution d'agréger les évaluations sachant que les utilisateurs n'ont pas tous le même niveau de crédibilité et certains peuvent avoir des identités virtuelles.

Différentes approches de confiance basées sur la crédibilité calculent la confiance comme une valeur numérique (ex., [44], [29], et [26]). Cependant, une valeur numérique ne permet de représenter ni l'incertitude sur les valeurs possibles de la confiance ni le manque de cohérence des évaluations. Ainsi, la valeur de la confiance obtenue est sujette à des interprétations ambiguës de la part des utilisateurs. La prise en compte de la crédibilité de l'utilisateur permet de résoudre le problème d'incohérence des évaluations (CF. Chapitre 3).

Les bases de données probabilistes associées à la sémantique de la théorie des mondes possibles constituent une solution intéressante au calcul de la confiance d'une ressource. Elles ont été largement utilisées pour représenter et analyser les données incertaines (extraction de données, etc.). Dans une base de données probabiliste chaque donnée (ou tuple) a une certaine probabilité d'appartenir à la base de données. Les bases de données probabilistes offrent alors une meilleure représentation de l'incertitude sous-jacente à la crédibilité des utilisateurs et permettent aussi à travers des requêtes un calcul incertain de la confiance d'une ressource [7].

Dans ce chapitre, nous proposons un modèle probabiliste d'évaluation de la confiance. Le modèle prend en entrée un ensemble d'évaluations sur une ressource donnée. Cet ensemble a préalablement subit un processus de filtrage afin de limiter la capacité d'attaque des utilisateurs Sybil et non crédibles (CF. Chapitre 4). Le modèle de confiance modélise sous forme d'une base de données proba-

bilistes selon différents modèles d'incertitudes introduits dans la littérature [7]. Le modèle calcule ensuite la confiance comme une évaluation de requêtes probabilistes.
Ce chapitre est organisé comme suit. La première section les notions de base des bases de données probabilistes. La deuxième et troisième section décrivent comment les évaluations sont structurées sous forme d'une base de données probabiliste en utilisant deux modèles d'incertitude. La derrière section définit notre méthode de calcul de la confiance sous forme d'évaluation de requêtes probabilistes.

5.2 Notions de base

Afin de faciliter la lecture de ce chapitre, nous introduisons tout d'abord les notions de base des bases de données probabilistes. Pour plus de détails, les lecteurs peuvent se référer à [3, 7].

5.2.1 Sémantique des probabilités dans une base de données

La sémantique des probabilités dans une base de données probabiliste n'est pas unique et diffère d'une application à une autre.

5.2.1.1 Sémantique des probabilités dans la littérature

Dans la plupart des cas, elle est liée à la manière dont les données sont extraites et pas nécessairement à la façon dont les données seront utilisées. Par exemple, les systèmes d'extraction d'information sont basés sur les modèles probabilistes et par conséquent les données extraites sont probabilistes [12]. De même, les méthodes d'analyse de données dans le but prévenir les crises financières s'appuient sur des modèles statistiques générant souvent des données probabilistes [17]. Dans de tels cas, les valeurs de probabilité possèdent une sémantique bien précise.

Dans d'autres cas, il n'y a pas de sémantique probabiliste mais seulement un niveau de fiabilité subjectif qui est transformé en une probabilité. Tel est le cas du nouvel outil de recherche Google Squared qui définit des nivaux de fiabilité (forte, faible, etc) des données. Les niveaux de probabilités sont par la suite transformés en des scores de probabilité associés aux données et enfin interrogés. Un autre exemple est le système Biorank permettant l'intégration de données scientifiques provenant de différentes sources afin de prédire des nouvelles fonctions de protéines à partir d'autres fonctions connues. Le système réalise un classement des fonctions de protéines de la plus probable à la moins probable.

5.2.1.2 Sémantique des probabilités dans notre modèle

Peu importe la façon dont elles sont dérivées, les probabilités représentent toujours un score de fiabilité dans l'intervalle [0, 1] [7]. La règle commune est qu'une plus grande valeur de probabilité représente toujours un degré plus élevé de confiance dans la validité de la donnée. Dans notre approche, nous avons évalué le niveau de confiance des évaluations par la crédibilité des utilisateurs. Nous associons alors la notion de crédibilité à celle de probabilité.
Prenons à titre d'exemple trois utilisateurs u_1, u_2, et u_3 ayant une expérience avec une ressource R_j et l'affirmation $\mathcal{S}$ suivante : u_i `a observé que` R_j `a satisfait à ses demandes`. Dans ce cas de figure l'incertitude reflète la probabilité que $\mathcal{S}$ se produise réellement. Cette probabilité peut être estimée en calculant la crédibilité (Cr_i) de l'utilisateur u_i. Soit trois événements e_1, e_2, et e_3 représentant respectivement le fait que u_1, u_2, et u_3 affirment chacun que R_j satisfait leurs demandes. Ces trois événements combinés lors du calcul de la confiance soulèvent des questions du type : compte tenu de la crédibilité variable des utilisateurs, comment agréger les évaluations et calculer la confiance d'une ressource à partir de leurs affirmations ?
Les bases de données probabilistes constituent une solution inté-

ressante au calcul de la confiance d'une ressource. En effet, les bases de données probabilistes offrent une meilleure représentation de l'incertitude sous-jacente à la crédibilité des utilisateurs et permettent aussi une évaluation des requêtes de calcul incertain de la confiance d'une ressource [7].

5.2.2 Types d'incertitudes dans une base de données probabiliste

Il existe deux types d'incertitude dans les bases de données probabilistes : l'incertitude au niveau du tuple et celle au niveau de l'attribut.
Dans la première, nous ne savons pas si le tuple appartient ou non à une instance de la base de données. Une variable aléatoire lui est alors associée. Le domaine de cette variable aléatoire est booléen. Elle est vraie lorsque le tuple est présent dans l'instance de la base de données et fausse lorsqu'il ne l'est pas. Un tel tuple est aussi appelé en anglais « maybe tuple »[47].
Dans la deuxième, la valeur d'un attribut est incertaine pour chaque tuple. Ce qui implique qu'à chaque attribut est associée une variable aléatoire et son domaine est l'ensemble de valeurs que l'attribut peut prendre.
Afin de simplifier le processus le traitement des requêtes, l'incertitude au niveau d'un attribut est souvent convertie en incertitude au niveau de tuple [47, 7]. Il suffit de créer pour chaque tuple t de la base, plusieurs clones t_1, t_2, ... où tous les attributs sont identiques sauf pour l'attribut incertain. Sa valeur dans chacun des clones correspondra à une des valeurs possibles qu'il peut prendre A_1, A_2, A_3... . A chaque tuple t_i est associée désormais une variable aléatoire.

5.2.3 Types de bases de données probabilistes

La première classification des bases de données probabilistes et la plus populaire identifie deux types : les bases de données tuples

indépendants et les bases de données blocs indépendants disjoints dites BID. Dans le premier type, tous les tuples sont des événements probabilistes indépendants. Dans le deuxième type, les tuples sont partitionnés en blocs de sorte que : (i) les tuples appartenant au même bloc sont des événements disjoints (mutuellement exclusifs,) et (ii) les tuples de différents blocs sont des événements indépendants. Ce type de modélisation permet de représenter les corrélations plus complexes pouvant exister entre les tuples de la base de données. La modélisation est similaire au processus de normalisation d'une base de données traditionnelle et consiste à décomposer la base de données en des composants indépendants et d'autres disjoints.
La deuxième classification des bases de données probabilistes concerne les bases de données où l'incertitude est au niveau des attributs. Deux types sont identifiés : bases de données probabilistes discrètes et bases de données probabilistes continues. Dans les premières, les attributs sont des variables aléatoires discrètes et dans les secondes, les attributs sont des variables aléatoires continues.

5.2.4 Définition formelle d'une base de données probabiliste

Formellement, une base de données probabiliste $\mathcal{BDP}rob$ est définie par un triplet ($\mathcal{S}$, $\mathcal{T}$, *prob*) où :

(i) $\mathcal{S}$ est le schéma d'une base données. Il est constitué d'un ensemble de relations probabilistes,

(ii) ($\mathcal{T}$) est un ensemble fini de tuples,

(iii) *prob* est une fonction associant une valeur de probabilité à chaque tuple $t \in \mathcal{T}$. $prob(t)$ représente la confiance dans l'existence du tuple dans la base de données. Plus la valeur de $prob(t)$ est élevée plus la confiance que t soit valide est élevée.

La modélisation des relations probabiliste est différente selon le modèle d'incertitude adopté : blocs indépendants ou tuples indépendants.

(a) **Relation probabiliste $\mathcal{BDP}rob$ tuples indépendants**. La relation probabiliste est de la forme $\mathcal{RP}rob(A_1,\ldots, A_k,p)$ où :

(a) $A_1,\ldots, A_k$ est un ensemble fini d'attributs,

(b) p est la probabilité associé à un tuple t d'une instance de la relation $\mathcal{RP}rob$.

La figure 5.1 montre l'allure générale d'une base de données probabiliste. Cette dernière constituée d'une seule table ayant n instances.

	A_1	...	A_k	p
t_1	...	...	...	p_1
...	...	...	...	...
t_n	...	...	...	p_n

FIGURE 5.1 – $\mathcal{BDP}rob$ tuples indépendants

(b) **Relation probabiliste $\mathcal{BDP}rob$ blocs indépendants**. La relation probabiliste est de la forme $\mathcal{RP}rob(A_1,\ldots, A_k,B_1,\ldots, B_m,p)$ où :

(i) $A_1,\ldots, A_k,B_1,\ldots, B_m$ est un ensemble fini d'attributs.

(ii) $A_1,\ldots, A_k$ sont les attributs identifiant de manière unique les blocs de tuples et représentent par conséquent la clé la base de données.

(iii) $B_1,\ldots, B_m$ sont les attributs incertains.

(iv) p est l'attribut probabilité.

La figure 5.2 montre l'allure générale d'une base de données probabiliste blocs indépendants. Cette dernière constituée d'une seule table ayant n composée de n blocs. Chaque bloc est constituée du même nombre de tuples m. Les blocs ont les mêmes valeurs pour les attributs $A_1, \ldots, A_k$ et des valeurs différentes pour les attributs $B_1, \ldots, B_m$.

	A_1	...	A_k	B_1	...	B_m	p
t_{11}	a_{11}	...	a_{k1}	b_{11}	...	b_{k1}	p_{11}
...				...	...	...	...
t_{1m}				b_{1m}	...	b_{km}	p_{1m}
	...	...	...	...	...	...	...
t_{n1}	a_{n1}	...	a_{k1}	b_{n1}	...	b_{k1}	p_{n1}
...				...	...	...	...
t_{nm}		...		b_{nm}	...	b_{km}	p_{nm}

FIGURE 5.2 – $\mathcal{BDP}rob$ blocs indépendants

5.2.5 Sémantique des mondes possibles

La $\mathcal{S}$émantique ($\mathcal{S}em$) de $\mathcal{BDP}rob$ est définie par le *modèle des mondes possibles* [7]. Dans [3] Cavallo et Pittarelli définissent $\mathcal{S}em(\mathcal{BDP}rob)$ par un espace de probabilité discret défini sur un nombre fini (n) d'instances d'une base de données. Les différents états de $\mathcal{BDP}rob$ sont appelés "mondes possibles " (mdp_k).
Une base de données $\mathcal{BDP}rob$ ayant n tuples peut avoir 2^n mondes possibles où chaque monde correspond à un sous ensemble de tuples. Les mondes possibles expriment l'incertitude suivante : "*L'un des mondes possibles sans savoir précisément lequel, est correct et les probabilités représentent les degrés de croyances dans les différents mondes possibles*" [15]. Chaque monde possible n'est que probable, en revanche tous les tuples d'un monde possible sont certains dans ce monde.
Formellement, $\mathcal{S}em(\mathcal{BDP}rob)$ est un couple ($\mathcal{MDP}$,$\mathcal{P}$) où :

(i) $\mathcal{MDP} = \{mdp_1, \ldots, mdp_n\}$,

(ii) $\mathcal{P}$: $\mathcal{MDP} \rightarrow [0, 1]$ de telle sorte que $\sum_{j=1,n} \mathcal{P}_j = 1$.

5.3 Vue d'ensemble du modèle probabiliste de confiance

Le modèle probabiliste de confiance prend en entrée l'ensemble des évaluations filtrées générées par le modèle de filtrage des éva-

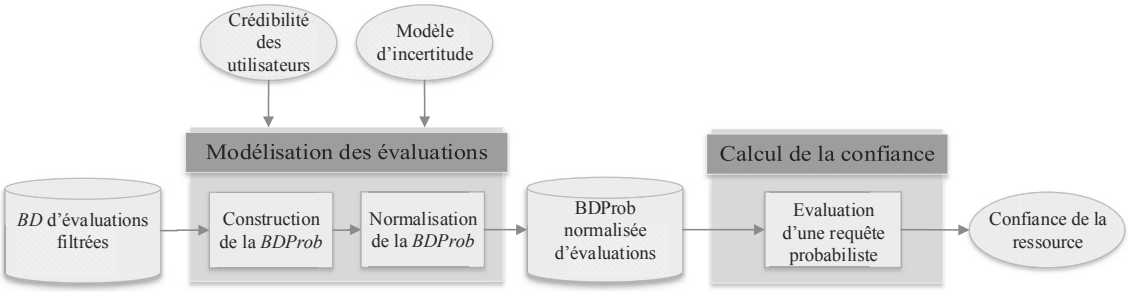


FIGURE 5.3 – Vue d'ensemble du modèle probabiliste de confiance

luations (CF. Chapitre 4). Après le choix du modèle de crédibilité et du modèle d'incertitude, les évaluations sont modélisées sous forme d'une base de données probabiliste. La modélisation se fait en deux étapes : la construction de la base de données probabiliste à partir d'une base de données traditionnelle et la normalisation. Ensuite, la confiance de la ressource est calculée en utilisant une requête probabiliste exécutée sur la base de données probabiliste d'évaluations normalisées (voir Figure 5.3).

5.4 Modélisation des évaluations avec un seul modèle de crédibilité

5.4.1 Construction de la base de données

Notre approche probabiliste vise à modéliser les évaluations collectées sous forme d'une base de données probabiliste $\mathcal{BDP}rob$ dans le but de calculer la confiance. Notre but est de créer des tuples t_i permettant le stockage de l'information suivante : `l'utilisateur` u_i `a fournit l'évaluation` X_i `de la ressource` R_j. Concrètement, t_i représente l'avis de u_i sur R_j. Or, dans notre approche cette information est considérée comme incertaine et nous l'avons évalué en calculant la crédibilité $\mathcal{CR}_i$ accordée à l'utilisateur. Notre définition de la crédibilité correspond alors à la définition de la pro-

babilité ($prob(t_i)$) de t_i dans une base de donnée probabiliste (CF. sous-section 5.2.1), à savoir un score de fiabilité dans l'interval [0, 1] attribué à t_i. Par conséquent, $prob(t)$ signifie à quel point cette information est correcte. Lorsque $prob(t)$ est égale à 1 (resp. 0) t_i est valide (resp. invalide) dans tous les mondes.
Afin de concevoir $\mathcal{BDP}rob$ nous traitons au préalable une $\mathcal{B}$ase de $\mathcal{D}$onnées ($\mathcal{BD}$) relationnelle traditionnelle contenant en plus des évaluations collectées, des informations supplémentaires sur les fournisseurs des ressources et la période d'évaluation. $\mathcal{BDP}rob$ est dérivée par extraction des avis pertinents de $\mathcal{BD}$ pour le calcul de la confiance.
Nous définissons le schéma relationnel de la base de données source $\mathcal{BD}$ par la relation $\mathcal{R}$(*ressource*, *utilisateur*, *évaluation*, Att_1, ..., Att_n) où :

(i) *ressource* est l'identifiant de R,

(ii) *utilisateur* désigne le nom de l'utilisateur,

(iii) *évaluation* représente le degré de satisfaction de l'utilisateur de cette ressource,

(iv) Att_1, ..., Att_n correspond à des détails supplémentaires tel que la date d'émission de l'évaluation ou des informations sur le fournisseur de la ressource.

Nous verrons par la suite comment nous exploitons les détails supplémentaires pour obtenir une évaluation précise de la confiance.
Nous construisons par la suite la base de données probabiliste ayant pour schéma $\mathcal{RP}rob$(*ressource*, *utilisateur*, *évaluation*, Att_1, ..., Att_n, p) qui correspond au même schéma de $\mathcal{BD}$ étendu par l'attribut probabilité p.

5.4.2 Illustration

En guise d'illustration, soit une base de données $\mathcal{BD}$ que nous transformons en une base de données probabiliste $\mathcal{BDP}rob$. $\mathcal{BD}$.

$\mathcal{BD}$ contient une relation probabiliste $\mathcal{R}(ressource, utilisateur, évaluation)$ où *ressource*, *utilisateur* et *évaluation* désignent respectivement l'identifiant de R, le nom de l'utilisateur et le degré de satisfaction de l'utilisateur de cette ressource (Fig. 5.4a).

Afin de simplifier la présentation de la base de données, nous représentons Att_1, ..., Att_n par une seule colonne marquée par .... Cette colonne est omise dans la représentation probabiliste de la base de données. $\mathcal{RP}rob$ est composée de trois tuples t_1, t_2, et t_3 ayant respectivement les probabilités 0.12, 0.84, et 0.88. Ces dernières correspondent aux valeurs de crédibilité calculées en utilisant notre modèle de crédibilité sur un ensemble aléatoire de données.

ressource	utilisateur	évaluation	...
R_1	u_1	0.2	
R_2	u_1	0.76	
R_1	u_3	0.97	

$\longrightarrow$

	ressource	utilisateur	évaluation	p
t_1	R_1	u_1	0.2	0.12
t_2	R_2	u_1	0.76	0.84
t_3	R_1	u_3	0.97	0.88

(a) $\mathcal{BD}$ *versus* $\mathcal{BDP}rob$

ressource	utilisateur	évaluation
R_1	u_1	0.2
R_2	u_1	0.76
R_1	u_3	0.97

$mdp_1, \mathcal{P}_1 = 0.09$

ressource	utilisateur	évaluation
R_1	u_1	0.2
R_2	u_1	0.76

$mdp_2, \mathcal{P}_2 = 0.1$

ressource	utilisateur	évaluation
R_1	u_1	0.2
R_1	u_3	0.97

$mdp_3, \mathcal{P}_3 = 0.02$

ressource	utilisateur	évaluation
R_2	u_1	0.76
R_1	u_3	0.97

$mdp_4, \mathcal{P}_4 = 0.65$

ressource	utilisateur	évaluation
R_1	u_1	0.2

$mdp_5, \mathcal{P}_5 = 0.01$

ressource	utilisateur	évaluation
R_2	u_1	0.76

$mdp_6, \mathcal{P}_6 = 0.09$

ressource	utilisateur	évaluation
R_1	u_3	0.97

$mdp_7, \mathcal{P}_7 = 0.12$

ressource	utilisateur	évaluation

$mdp_8, \mathcal{P}_8 = 0.12$

(b) les mondes possibles de $\mathcal{BDP}rob$

FIGURE 5.4 – Illustration d'une base de données probabiliste

La figure 5.4b présente les mondes possibles mdp_k de $\mathcal{BDP}rob$ et

les probabilités correspondantes ($\mathcal{P}_k$). Chaque mdp_k contient un sous-ensemble de tuples présents dans $\mathcal{BDP}rob$. La probabilité $\mathcal{P}_k$ est calculée en utilisant l'hypothèse de l'indépendance des tuples de la façon suivante : nous multiplions les probabilités d'existence des tuples présents dans mdp_k par les probabilités de la non-existence des autres tuples absents de mdp_k. Par exemple, la probabilité $\mathcal{P}_2$ du monde $mdp_2=\{t_1, t_2\}$ est calculée de la sorte : 0.12*0.84*(1-0.88) = 0.01.

L'interprétation des mondes possibles est extrêmement intuitive et offre une sémantique claire pour l'évaluation des requêtes sur les bases de données probabilistes. Soit $\Phi_{ressource=R_1}$ une requête recherchant R_1 dans certains tuples de $\mathcal{BDP}rob$. Cette requête est évaluée sur chaque monde possible mdp_k séparément ($\Phi(mdp_k)$). La probabilité associée à $\Phi(mdp_k)$ est $\mathcal{P}_k$. Le résultat final de la requête est une agrégation des résultats intermédiaires obtenus dans les différents mondes possibles et correspond à un ensemble de tuples t' : $\cup_{k=1,8}\Phi(mdp_k)$. L'équation 5.1 calcule $\mathcal{P}rob(t')$ de la sorte :

$$\mathcal{P}rob(t') = \sum_{k,\ t' \in \Phi(mdp_k)} \mathcal{P}_k \qquad (5.1)$$

La figure 5.5a montre le résultat de l'exécution de $\Phi_{ressource=R_1}$ sur $mdp_{k=1,8}$. Le résultat sur $\Phi(mdp_6)$ et $\Phi(mdp_8)$ est un ensemble vide mais avec des probabilités non nulles. Bien que l'ensemble soit vide, cette information peut être pertinente pour un utilisateur pour indiquer une insuffisance de données existantes afin de trouver des résultats pertinents.

La figure 5.5b montre le calcul final de la probabilité. L'inexactitude des données donne lieu à un grand nombre de résultats avec des probabilités faibles par conséquent de faible précision. Cependant, les utilisateurs apprécieraient des résultats avec des probabilités élevées.

ressource	utilisateur	évaluation
R_1	u_1	0.2
R_1	u_3	0.97

$\Phi(mdp_1), \mathcal{P}_1 = 0.09$

ressource	utilisateur	évaluation
R_1	u_1	0.2

$\Phi(mdp_2), \mathcal{P}_2 = 0.1$

ressource	utilisateur	évaluation
R_1	u_1	0.2
R_1	u_3	0.97

$\Phi(mdp_3), \mathcal{P}_3 = 0.02$

ressource	utilisateur	évaluation
R_1	u_3	0.97

$\Phi(mdp_4), \mathcal{P}_4 = 0.65$

ressource	utilisateur	évaluation
R_1	u_1	0.2

$\Phi(mdp_5), \mathcal{P}_5 = 0.01$

ressource	utilisateur	évaluation

$\Phi(mdp_6), \mathcal{P}_6 = 0.09$

ressource	utilisateur	évaluation
R_1	u_3	0.97

$\Phi(mdp_7), \mathcal{P}_7 = 0.12$

ressource	utilisateur	évaluation

$\Phi(mdp_8), \mathcal{P}_8 = 0.12$

(a) Mondes possibles résultants

ressource	utilisateur	évaluation	Probabilités
R_1	u_1	0.2	$\mathcal{P}_1+\mathcal{P}_2+\mathcal{P}_3+\mathcal{P}_5$=0.22
R_1	u_3	0.97	$\mathcal{P}_1+\mathcal{P}_3+\mathcal{P}_4+\mathcal{P}_7$=0.88
			$\mathcal{P}_6+\mathcal{P}_8$=0.21

(b) Résultat de la requête

FIGURE 5.5 – Evaluation d'un requête sur $\mathcal{BDP}rob$

5.4.3 Normalisation de la base de données

Notons que $\mathcal{RP}rob$ contient des tuples rattachés à des utilisateurs fournissant des évaluations sur différentes ressources. Ces utilisateurs peuvent être cohérents (toujours crédibles ou non) ou inconsistants (i.e., basculent entre des comportements crédibles et non crédibles) dans leurs évaluations. En effet, certains utilisateurs sont plus crédibles que d'autres en fournissant des évaluations correctes, alors que d'autres, en faisant l'inverse, sont moins crédibles. Considérons deux tuples t_1 et t_2 correspondant à des évaluations fournies par u_1. Si t_1 est faux, alors il est faux parce que u_1 n'est pas crédible. Il est aussi probable que t_2 soit faux. Par conséquent, si un tuple est faux la probabilité que l'autre tuple correspondant au même utilisateur soit également faux se voit accroître. Ainsi, le

modèle de données probabilistes proposé ne respecte pas le modèle de tuples indépendants (ex., [7]) pour lequel une probabilité est associée à chaque tuple indépendamment de l'occurrence des autres tuples.

Représenter une base de données probabilistes dans laquelle tous les tuples sont indépendants s'avère être une tâche délicate. Cependant, des bases de données plus complexes peuvent parfois être décomposées en des relations de tuples indépendants et ensuite être normalisées [39].

La figure 5.6 montre comment nous normalisons $\mathcal{BDP}rob$ ($\mathcal{BDP}rob^{\mathcal{N}}$) en deux relations probabilistes de tuples indépendants : $\mathcal{PAIR}$ et $\mathcal{RP}rob_1$. $\mathcal{PAIR}$ stocke tous les utilisateurs avec leur valeur respective de crédibilité. Vu que $\mathcal{PAIR}$ doit être souvent mis à jour nous la traitons comme une vue plutôt qu'une table. L'utilisateur u_i est crédible à propos de R_j si ses évaluations sont consistantes. L'équation 5.2 calcule la crédibilité de u_i ($\mathcal{CR}_i$) sur la base des évaluations fournies dans le passé.

$$\mathcal{CR}_i = \prod_j \mathcal{CR}_i^j \tag{5.2}$$

A partir de $\mathcal{RP}rob$, nous calculons $\mathcal{CR}_1$ comme $0.12 * 0.84 = 0.1$. Comme u_3 fournit une seule évaluation, $\mathcal{CR}_3$ reste la même dans $\mathcal{PAIR}$. $\mathcal{RP}rob_1$ stocke tous les tuples maintenant indépendants de la crédibilité de l'utilisateur.

$\mathcal{PAIR}$

utilisateur	p
u_1	0.1
u_3	0.88

$\mathcal{RP}rob_1$

ressource	utilisateur	évaluation	p
R_1	u_1	0.2	0.12
R_2	u_1	0.76	0.84
R_1	u_3	0.97	0.88

FIGURE 5.6 – Normalisation de $\mathcal{BDP}rob$

5.5 Modélisation des évaluations avec plusieurs modèles de crédibilité

Cette section explique comment nous modélisons les évaluations avec plusieurs modèles de crédibilité en utilisant une base de données blocs indépendants.

5.5.1 Construction de la base de données

Plusieurs solutions existent pour l'évaluation de la crédibilité de l'utilisateur. Il s'agit essentiellement de trouver des critères d'évaluation de l'honnêteté de l'utilisateur. Généralement, ces critères sont plutôt complémentaires que contradictoires. Comme par exemple le critère d'expertise dans le modèle proposé dans [33] et celui de fiabilité proposé dans [26, 29, 46, 45]. Il serait alors intéressant de pouvoir profiter des avantages de différents modèles de crédibilités à la fois dans l'évaluation d'une unique ressource R. L'information à stocker dans la base de données est inchangée par rapport à celle utilisée dans la sous section 5.4, à savoir des tuples t_i représentant chacune l'avis d'un utilisateur u_i sur la ressource R. La différence, cette fois-ci est que plusieurs modèles de crédibilités sont utilisées pour le calcul de l'incertitude de cette information.
Il possible de représenter ce type d'information en utilisant le modèle d'incertitude blocs indépendants. Nous créons alors pour chaque tuple t_{ij} de clones pour chaque modèle de crédibilité j. La probabilité ($prob_j(t_i)$) de t_{ij} change selon le modèle et correspond à la valeur de crédibilité Cr_{ji} calculée par le modèle M_j comme le montre la figure 5.8.
Afin de concevoir la base de données probabiliste blocs indépendants $\mathcal{BIDP}rob$ nous utilisons la même base de données traditionnelle source $\mathcal{BD}$ utilisée dans la sous section précédente ayant pour schéma $\mathcal{R}$(ressource, utilisateur, évaluation, Att_1, ..., Att_n). Nous construisons $\mathcal{BIDP}rob$ ayant pour schéma $\mathcal{R}'\mathcal{P}rob$(ressource, utilisateur, évaluation,

modèle, Att_1, ..., Att_n, p) qui correspond au même schéma de $\mathcal{BD}$ étendu par l'attribut modèle désignant le modèle de crédibilité et l'attribut probabilité p. Les attributs ressource, utilisateur et évaluation représentent les attributs clé identifiant de manière unique chaque bloc.
Vérifions maintenant que ce modèle de stockage des données correspond au modèle de base de données probabiliste blocs indépendants. Il suffit de séparer les tuples données par le même utilisateur dans des blocs différents. Les tuples appartenant à un même bloc possèdent tous la même information sauf pour le choix du modèle de crédibilité. Or le choix du modèle de crédibilité est exclusif puisque chacun de ces modèles est basé sur une théorie différente. Nous pouvons alors conclure que les tuples d'un même bloc correspondent à des événements disjoints ou mutuellement exclusifs. Notre modélisation valide alors la première règle de la modélisation des bases de données BID. La deuxième règle est également valide puisque les tuples appartenant à des blocs différents représentent des avis fournis par différents utilisateurs et sont par conséquent indépendants.

5.5.2 Illustration

Reprenons l'exemple de la base de données $\mathcal{BD}$ utilisé dans la section précédente. Cette fois, nous transformons $\mathcal{BD}$ en une base de données probabiliste $\mathcal{BIDProb}$ ayant pour relation probabiliste $\mathcal{R'Prob}$ (ressource, utilisateur, évaluation, modèle, p) (illustration dans la figure 5.8). Afin de visualiser les blocs d'une manière plus claire, nous fusionnons les cellules de chaque attribut clé de chaque bloc puisque ces attributs ne changent pas de valeurs dans le même bloc.
Afin de générer les tuples de $\mathcal{BIDProb}$ nous utilisons deux modèles de crédibilité : le premier basé sur le clustering $\mathcal{K}$-means ($\mathcal{M}_1$) et le deuxième basé sur le clustering C-means ($\mathcal{M}_2$) (voir Chapitre 3). Selon $\mathcal{M}_1$ les valeurs de crédibilité des utilisateurs u_1, u_2 et u_3

sont respectivement 0.14, 0.88 et 0.86 et selon $\mathcal{M}_2$ elles sont 0.12, 0.84, et 0.88. Afin de générer les probabilités des tuples, nous multiplions chaque probabilité par la probabilité que le modèle $\mathcal{M}_i$ soit utilisé qui correspond à 0.5. Puisque chaque modèle a une chance sur deux d'être utilisé. Les probabilités des tuples t_{11}, t_{21} et t_{31} ayant pour valeur d'attribut modèle $\mathcal{M}_1$ sont respectivement 0.07, 0.44 et 0.43 et les tuples t_{12}, t_{22} et t_{32} ayant pour valeur d'attribut modèle $\mathcal{M}_2$ sont respectivement 0.06, 0.42 et 0.44.

ressource	utilisateur	évaluation	...
R_1	u_1	0.2	
R_1	u_2	0.76	
R_1	u_3	0.97	

$\mathcal{BD}$

$\longrightarrow$

	ressource	utilisateur	évaluation	modèle	probabilité
t_{11}	R_1	u_1	0,2	M_1	0.07
t_{12}				M_2	0.06
t_{21}	R_1	u_2	0,76	M_1	0.44
t_{22}				M_2	0.42
t_{31}	R_1	u_3	0,97	M_1	0.43
t_{32}				M_2	0.44

$\mathcal{BIDProb}$

FIGURE 5.7 – *$\mathcal{BD}$ versus $\mathcal{BIDProb}$*

La figure 5.9 montre les mondes possibles de la base de données $\mathcal{BIDProb}$. Dans notre approche, nous proposons deux techniques pour la génération des mondes possibles. Dans la première technique, nous considérons que les modèles de crédibilité $\mathcal{M}_\infty$ et $\mathcal{M}_\in$ ne présentent pas corrélation. Il est alors possible d'utiliser les deux $\mathcal{M}_\infty$ et $\mathcal{M}_\in$ en même temps. Dans la deuxième technique, les modèles de crédibilité $\mathcal{M}_\infty$ et $\mathcal{M}_\in$ présentent une corrélation et donc un seul modèle est utilisé à la fois pour la génération des mondes possibles.

A notre connaissance, l'évaluation des requêtes probabilistes des bases de données probabilistes est limitée et en particulier pour les opérateurs d'agrégation. Nous nous limitons alors au modèle de données tuples indépendants pour le calcul de la valeur de confiance.

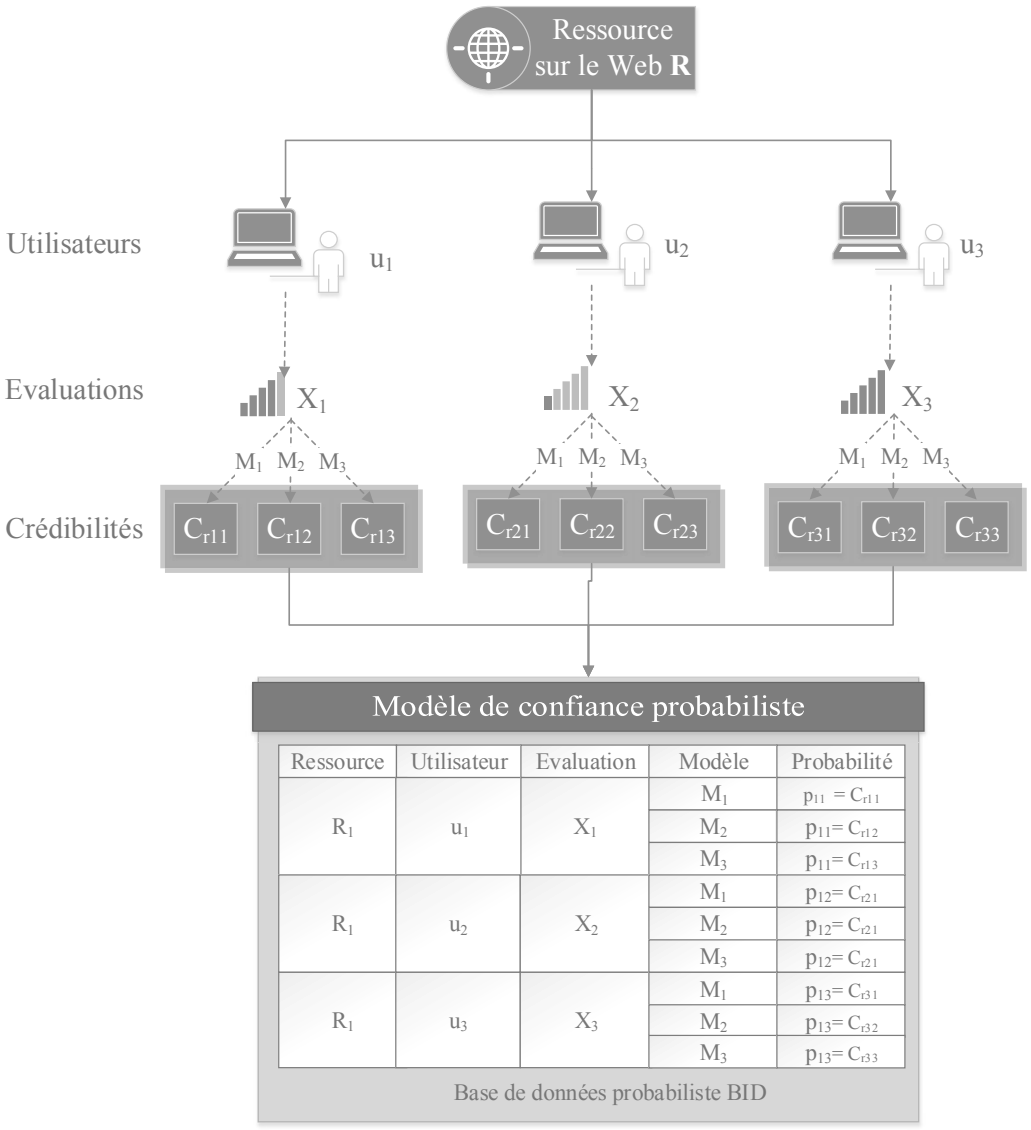


FIGURE 5.8 – Génération de la base de données BID

Monde possible	Probabilité
$W'^1 = \{t'_{11}, t'_{21}, t'_{31}\}$	0,013
$W'^2 = \{t'_{11}, t'_{21}, t'_{32}\}$	0,013
$W'^3 = \{t'_{11}, t'_{22}, t'_{31}\}$	0,013
$W'^4 = \{t'_{11}, t'_{22}, t'_{32}\}$	0,013
$W'^5 = \{t'_{12}, t'_{21}, t'_{31}\}$	0,011
$W'^6 = \{t'_{12}, t'_{21}, t'_{32}\}$	0,011
$W'^7 = \{t'_{12}, t'_{22}, t'_{31}\}$	0,011
$W'^8 = \{t'_{12}, t'_{22}, t'_{32}\}$	0,011
$W'^9 = \{t'_{11}, t'_{21}\}$	0,004
$W'^{10} = \{t'_{11}, t'_{22}\}$	0,004
$W'^{11} = \{t'_{11}, t'_{31}\}$	0,004
$W'^{12} = \{t'_{11}, t'_{32}\}$	0,004
$W'^{13} = \{t'_{12}, t'_{21}\}$	0,003
$W'^{14} = \{t'_{12}, t'_{22}\}$	0,003
$W'^{15} = \{t'_{12}, t'_{31}\}$	0,004
$W'^{16} = \{t'_{12}, t'_{32}\}$	0,004
$W'^{17} = \{t'_{21}, t'_{31}\}$	0,163
$W'^{18} = \{t'_{21}, t'_{32}\}$	0,025
$W'^{19} = \{t'_{22}, t'_{31}\}$	0,157
$W'^{20} = \{t'_{22}, t'_{32}\}$	0,161
$W'^{21} = \{t'_{11}\}$	0,001
$W'^{22} = \{t'_{12}\}$	0,001
$W'^{23} = \{t'_{21}\}$	0,049
$W'^{24} = \{t'_{22}\}$	0,048
$W'^{25} = \{t'_{31}\}$	0,054
$W'^{26} = \{t'_{32}\}$	0,05
$W'^{27} = \{\oslash\}$	0,016

(a) Mondes possibles sans corrélation

Monde possible	Modèle	Probabilité
$W^1 = \{t_{11}, t_{21}, t_{31}\}$	M_1	0,0131
$W^2 = \{t_{11}, t_{21}\}$	M_1	0,004
$W^3 = \{t_{11}, t_{31}\}$	M_1	0,0044
$W^4 = \{t_{21}, t_{31}\}$	M_1	0,1627
$W^5 = \{t_{11}\}$	M_1	0,0045
$W^6 = \{t_{21}\}$	M_1	0,0305
$W^7 = \{t_{31}\}$	M_1	0,0341
$W^8 = \{t_{12}, t_{22}, t_{32}\}$	M_2	0,0111
$W^9 = \{t_{12}, t_{22}\}$	M_2	0,0033
$W^{10} = \{t_{12}, t_{32}\}$	M_2	0,0038
$W^{11} = \{t_{22}, t_{32}\}$	M_2	0,1608
$W^{12} = \{t_{12}\}$	M_2	0,0011
$W^{13} = \{t_{22}\}$	M_2	0,0475
$W^{14} = \{t_{32}\}$	M_2	0,0498
$W^{15} = \{\oslash\}$		0,0164

(b) Mondes possibles avec corrélation

FIGURE 5.9 – Mondes possibles de $\mathcal{BIDProb}$

5.6 Calcul de la confiance

Pour établir la confiance d'un ressource sur le Web à partir de $\mathcal{BDP}rob^{\mathcal{N}}$, nous développons des requêtes spécifiques. Dans cette section, nous expliquons le choix de la requête ainsi que le principe de l'évaluation de cette dernière.

5.6.1 Choix de la requête

Etablir la confiance d'une ressource consiste à agréger les évaluations des utilisateurs en une valeur probabiliste. Ceci peut être exprimé à l'aide d'une requête SQL $SELECT\ AVG$ pour obtenir l'agrégation des évaluations à partir de $\mathcal{RP}rob_1$. Intuitivement, l'application de cette requête sur mdp_k signifie que les utilisateurs dans mdp_k remarquent **conjointement** que R_j satisfait leurs demandes avec une probabilité $\mathcal{P}_k$. La requête la plus simple permettant le calcul de la confiance est la requête SQL $\mathcal{F}_{AVG(e)}(\sigma_{r=R_j})$ suivante :

SELECT AVG(*évaluation*) **FROM** $\mathcal{BDP}rob^{\mathcal{N}}$ **WHERE** $ressource = R_j$;

$\mathcal{BDP}rob^{\mathcal{N}}$ s'interprète comme $2^5 = 32$ mondes possibles mdp_k. La figure 5.10a illustre le contenu du mdp_1. La figure 5.10b elle montre le résultat de l'évaluation de $\mathcal{F}_{AVG(e)}(\sigma_{r=R_1})$ comme un ensemble de quatre réponses possibles pour la valeur de confiance (ex., 0.97, 0.585, $\emptyset$, et 0.2), ordonnées selon leur probabilité d'existence. Dans [18] Jayram et al. présentent le résultat de $\mathcal{F}_{AVG()}(\sigma)$ sur les bases de données probabilistes comme une moyenne pondérée des réponses possibles de la valeur de la confiance.
Contrairement aux approches probabilistes existantes (p.ex., [41] et [54]), nous proposons des requêtes SQL personnalisées en fonction des préférences des utilisateurs pour le calcul de la confiance. De plus, plusieurs types d'information peuvent être extraites de

	utilisateur
t_1	u_1
t_2	u_3

	ressource	utilisateur	évaluation
t'_1	R_1	u_1	0.2
t'_2	R_2	u_1	0.76
t'_3	R_1	u_3	0.97

(a) mdp_1, $\mathcal{P}_1$=0.008

valeur de la confiance	p
0.970	0.774
0.585	0.106
$\emptyset$	0.106
0.20	0.014

(b) Résultats de la requête

FIGURE 5.10 – Evaluation de la requête $\mathcal{F}_{AVG(e)}(\sigma_{r=R_1})$ sur $\mathcal{BDP}rob^{\mathcal{N}}$

$\mathcal{BDP}rob^{\mathcal{N}}$. Nous listons, ci-après, quatre variantes de requêtes de calcul de la confiance :

1. Q_1 retourne la confiance de R_j sous forme d'une moyenne des évaluations fournies sur R_j :
   Q_1 : **SELECT** **AVG**(*évaluation*) **FROM** $\mathcal{BDP}rob^{\mathcal{N}}$
   WHERE $ressource = R_j$;

2. Q_2 considère qu'il existe une liste prédéfinie $\{u_1, \ldots, u_k\}$ d'utilisateurs crédibles connus a priori et calcule la confiance comme une moyenne de leurs évaluations :
   Q_2 : **SELECT** **AVG**(*évaluation*) **FROM** $\mathcal{BDP}rob^{\mathcal{N}}$
   WHERE $ressource = R_j$ **AND** $utilisateur\ IN\ (u_1, \ldots, u_k)$;

3. Q_3 retourne la confiance de R_j comme une moyenne des évaluations fournies sur cette ressource à partir d'une date donnée :
   Q_3 : **SELECT** **AVG**(*évaluation*) **FROM** $\mathcal{BDP}rob^{\mathcal{N}}$
   WHERE $ressource = R_j$ **AND** $date \geq$ "2014 − 01 − 01";

4. Q_4 retourne la confiance d'un fournisseur ($fourn_i$) comme une moyenne des évaluations fournies sur tous les ressources de $fourn_i$:
   Q_4 : **SELECT** **AVG**(*évaluation*) **FROM** $\mathcal{BDP}rob^{\mathcal{N}}$
   WHERE $fournisseur = fourn_i$;

5.6.2 Principe d'évaluation de la requête

Malgré la simplicité de la sémantique des mondes possibles, ils suscitent des sérieux problèmes de calcul même pour requêtes les plus simples [7]. Plusieurs études ont démontré que le problème d'évaluation d'une requête est $\mathcal{P}$-difficile quelque soit le modèle de données probabiliste utilisé. La complexité du problème est exponentielle par rapport au nombre de tuples de la base de données. Ainsi, des algorithmes (p.ex., [7] et [18]) ont proposés pour traiter les requêtes complexes sur des flux de données massifs. Dans [18] Jayram et al. proposent des algorithmes qui se basent sur des fonctions mathématiques permettant de calculer des approximations des opérateurs d'agrégation globaux ($\mathcal{SUM}, \mathcal{COUNT}, \mathcal{AVG}$) très proches des valeurs réelles.
Dans ce travail, nous adoptons l'algorithme de Jayram et al. [18] relatif au calcul de l'opérateur $\mathcal{AVG}$ pour le calcul de la confiance des ressources. Les auteurs proposent une fonction mathématique le problème de l'estimation d'$\mathcal{AVG}$. Le problème est donc réduit à comment trouver estimation efficace de l'intégrale d'une fonction. Cette réduction permet d'obtenir un algorithme exact de complexité $O(nlog^2n)$. Notre choix a été guidé par les exigences suivantes : (i) une bonne approximation de l'opérateur $\mathcal{AVG}$; (ii) une faible complexité de l'algorithme pour assurer une évaluation performante de la confiance (comparé à une complexité de $O(n^3)$) dans [] BDJ05) et enfin (iii) le modèle de données correspond au modèle d'incertitude que nous avons adopté à savoir le modèle de données tuples indépendants.

5.6.3 Algorithme d'évaluation de la requête

L'estimation d'AVG correspond à l'intégrale de la fonction mathématique $h_{AVG}(x)$ définie par le théorème 1 [18]. La fonction est basée la notion de flux de données probabiliste qu'ils définissent comme " *une séquence de n tuples incertains* $(t_i, p(t_i))$ *où chaque*

tuple est présent dans une instance de la base avec une probabilité $p(t_i) \in (0, 1]$ *(indépendamment de tous les autres tuples dans la base de données)*".

Théorème 1 *[18] Pour le flux probabiliste,* $F = (V_1, ..., V_n)$ *Soit*

$$h_{AVG}(x) = \sum_i \mathcal{X}_i.p(t_i). \prod_{j \neq i}(1 - p(t_j) + p(t_j)x).$$

$$Alors,\ AVG = \int_0^1 h_{AVG}\, \mathrm{d}x.$$

Nous proposons un algorithme pour le calcul d'une certaine ressource Web R_j. L'algorithme prend en entrée la base de données probabiliste normalisée $\mathcal{BDP}rob^{\mathcal{N}}$ et la ressource R_j à évaluer. L'algorithme opère en deux étapes. La première étape consiste à extraire l'ensemble des évaluations correspondants à la ressource R_j. La deuxième étape calcul l'estimation de la valeur de la confiance selon le théorème 1.

Algorithme 11 : Calcul de la confiance

Entrée : $\mathcal{BDP}rob^{\mathcal{N}}$, R_j

Sortie : la confiance$C$

$\mathcal{X} \leftarrow$ **SELECT** (*évaluation*) **FROM** $\mathcal{BDP}rob^{\mathcal{N}}$ **WHERE** $ressource = R_j$;

$h_{AVG}(x) \leftarrow \sum_i \mathcal{X}_i.p(t_i). \prod_{j \neq i}(1 - p(t_j) + p(t_j)x)$

C ← estimerIntégrale (0, 1, AVG(x))

return C

5.7 Conclusion

Dans ce chapitre, nous avons proposé un modèle probabiliste d'évaluation de la confiance des ressources sur le Web. Le modèle s'appuie sur la théorie des bases de données probabilistes et la sémantique des mondes possibles.

Chapitre 6

Étude expérimentale

Sommaire

6.1 Introduction

Les systèmes de confiance sont confrontés à des problèmes très complexes et assez difficiles à résoudre compromettant leur performance (CF. Chapitre 2 section 2.1.2). En plus du critère de la performance, les systèmes de confiance doivent être robustes contre les attaques externes afin de permettre leur intégration dans les applications distribuées à grande échelle.

Dans ce chapitre, nous proposons la mise en oeuvre de notre approche d'évaluation des confiance des ressources sur le Web afin de valider son efficacité et sa robustesse.

L'approche d'évaluation de confiance est basée sur trois modèles : le modèle de crédibilité flou, le modèle de filtrage des évaluations et le modèle probabiliste de confiance (CF. chapitres 3, 4 et 5). Nous développons alors un système d'évaluation de la confiance des ressources sur le Web basé sur les trois modèles que nous avons appelé WRTrust. Nous évaluons ensuite la performance et la robustesse de ce système en réalisant différentes expérimentations sur les différents modèles. Pour déstabiliser le système, nous injections des évaluations invalides données par des utilisateurs malicieux. Ces évaluations nous permettent de simuler les attaques des évaluations biaisées et les attaques Sybil. La performance du système est évaluée en termes de la qualité de la valeur de la confiance obtenue. Quand à la robustesse, elle se définit par la stabilité de la performance du système lorsque qu'il est est soumis à des sollicitations inhabituelles. Dans notre cas, nous analysant la robustesse de notre système de confiance en simulant les attaques d'évaluations biaisées et les attaques Sybil.

Ce chapitre comporte trois sections. La première, décrit notre système d'évaluation de la confiance. La deuxième, les différentes expérimentations réalisées : les jeux de données, le paramétrage, les métriques et enfin les résultats obtenus.

6.2 Système d'évaluation de la confiance WRTrust

WRTrust est un système d'évaluation des ressources sur le Web. Nous présentons dans cette section :

6.2.1 Description de WRTrust

WRTrust comprend quatre composants principaux : *Collecteur d'évaluation et de confiance*, *Evaluateur de crédibilité*, *Filtreur des évaluations*, et *Evaluateur de confiance*. Pour des raisons de performance, nous suggérons d'héberger ces composants du côté client. La figure 6.1 illustre un environnement supportant l'exécution de notre système distribuée d'évaluation de la confiance.

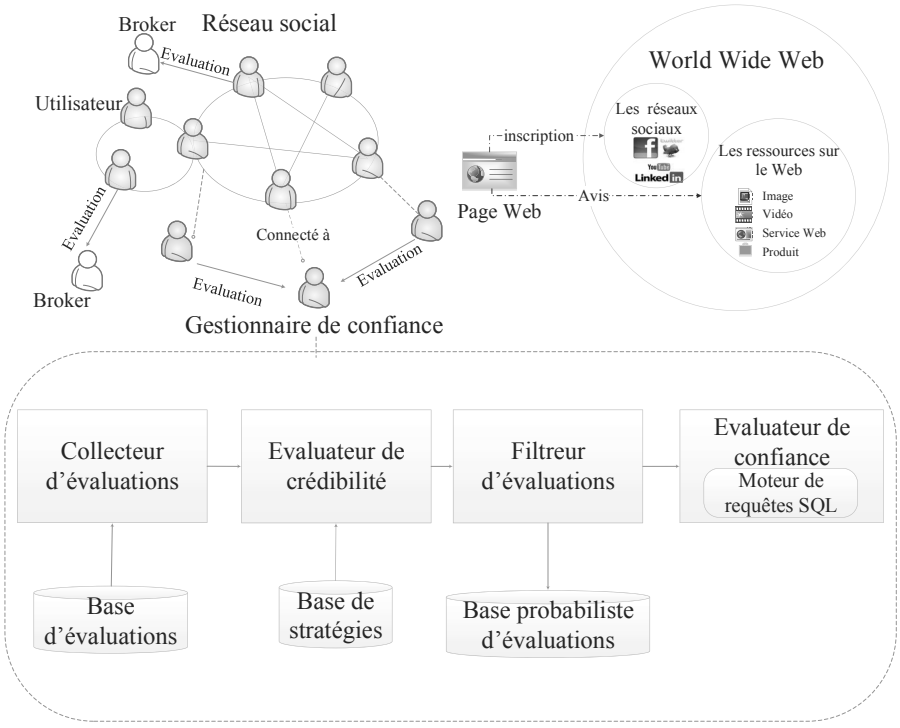


FIGURE 6.1 – Système d'évaluation de la confiance

Lors de la souscription au système, des *gestionnaires de confiance* sont déployés sur les plates-formes des utilisateurs. Après chaque transaction, l'utilisateur envoie à son *gestionnaire de confiance* une

évaluation relative à un retour d'expérience avec la ressource. Ces évaluations sont stockées dans la base de données d'évaluations. Un utilisateur potentiel interroge son *gestionnaire de confiance* sur la fiabilité de certaines ressources. Dès réception de cette demande, le *Collecteur d'évaluation et de confiance* collecte un ensemble d'évaluations fournies par des utilisateurs d'une même communauté provenant de différents réseaux sociaux ou bien par d'autres *Gestionnaires de confiance*. L'*Évaluateur de crédibilité* calcule ensuite la crédibilité des utilisateurs selon le modèle de crédibilité flou (CF. chapitre 3). Le *Filtreur des évaluations* filtre les évaluations selon le modèle de filtrage (CF. Chapitre 4) afin de ne garder que les évaluations des utilisateurs crédibles et non Sybil. Selon le modèle probabiliste (CF. chapitre 5), une base de données probabilistes est générée en fonction des valeurs de crédibilité des utilisateurs. Dans cette base de données, une valeur de probabilité est associée à chaque tuple. Elle représente la valeur de crédibilité de l'utilisateur fournissant cette évaluation. L'*Évaluateur de Confiance* calcule la confiance déterministe (CF. Chapitre 3) en utilisant des évaluations des autres utilisateurs. Lorsque l'utilisateur a des exigences de sécurité stricts, l'*Evaluateur de Confiance* calcule la confiance probabiliste (CF. chapitre 5) offrant aux utilisateurs une analyse de risque plus raffinée. L'*Evaluateur de Confiance* stocke les informations sur la confiance dans une base de données partagée avec les autres utilisateurs de la même communauté.
Enfin, l'*Evaluateur de Confiance* calcule la confiance à travers une requête probabiliste exécuté sur cette base. Finalement, le *Gestionnaire de confiance* renvoie à l'utilisateur la ressource la plus fiable.

6.2.2 Description détaillée des composants de WRTrust

Nous décrivons dans ce qui suit de manière détaillée les composants du système WRTrust.
Le **Collecteur d'évaluation et de confiance** prend en charge les requêtes sur les valeurs de confiance de la part des utilisateurs po-

tentiels dans un contexte donné. Dans notre approche, nous collectons les évaluations ainsi que les valeurs de confiance. Afin de rendre l'information collectée disponible à la demande, des *brokers* sont déployés sur divers réseaux sociaux. Ces derniers mettent à jour dynamiquement les évaluations à la fin de chaque transaction et les valeurs de confiance après chaque demande de calcul de la confiance. Ainsi, lorsque les valeurs de confiance sont collectées à la demande, l'utilisateur reçoit plus rapidement la réponse de la part du *gestionnaire de confiance* en comparaison avec d'autres approches telles que [26] et [29]. De plus, l'utilisateur est informé de toute transaction et/ou demande de calcul de la confiance.

L'**Evaluateur de crédibilité** vérifie si toutes les évaluations fournies par des utilisateurs sans *gestionnaire de confiance* ou d'autres *gestionnaires de confiance* sont valides. Pour ce faire, l'*Evaluateur de crédibilité* examine le répertoire des évaluations pour rechercher celles provenant de pairs ayant les liens sociaux tels que l'amitié et la supervision avec l'utilisateur potentiel. L'avis de la majorité est construit en utilisant les stratégies définies par notre modèle de crédibilité et selon les caractéristiques des clusters flous (ex., nombre d'utilisateurs ayant des degrés d'appartenance « faible » et/ou « fort »). Ainsi, un utilisateur doit spécifier ses propres valeurs floues « faible » et « fort » comme étant des fonctions d'appartenance dépendant du contexte de la ressource (ex., critique ou pas).

L'**Evaluateur de confiance** est un outil permettant aux utilisateurs de définir leurs préférences pour spécifier la confiance. Certaines requêtes nécessitent de la part de l'*Evaluateur de confiance* une évaluation de la confiance avec des contraintes telles que des périodes d'évaluation spécifiques ou uniquement l'utilisation des évaluations.

6.2.3 Description technique

Nous avons mis en oeuvre le système WRTrust en deux étapes. La première est l'implémentation et des tests des différents algorithmes proposés dans chaque modèle. La deuxième est l'implémentation d'une application Web pour l'évaluation de la confiance des ressources sur le Web (voir Annexe 7.3).

6.2.3.1 Choix technologiques

Nous avons opté pour les choix technologiques suivants :

(i) Technologies côté serveur : nous avons utilisé le langage JAVA pour implémenter des différentes algorithmes et PostgreSQL pour stocker les évaluations et estimer la confiance des ressources sur le Web,

(ii) Technologies du côté client : nous avons utilisé l'outil Bootstrap et le langage JavaScript pour la réalisation des pages Web,

(iii) Outils et bibliothèques : nous avons utilisé l'IDE Netbeans, la bibliothèque *Apache Mahout* qui offre une implémentation de l'algorithme flou *C*-means,

6.3 Protocoles d'expérimentation

Cette section décrit les différentes expérimentations réalisées.

6.3.1 Jeux de données

Il existe une multitude de jeux de données accessibles sur le Web pour l'évaluation de différents types de ressources. Nous avons choisi comme domaine d'application de notre approche : les films et les services Web.

MovieLens [1] MovieLens est un le jeu de données d'évaluations de films. Nous avons opté pour ce jeu de données pour sa simplicité et aussi parce qu'il fournit des données réelles. Ce dernier

contient 100000 évaluations données par 943 utilisateurs sur 1682 films. Les évaluations représentent le taux de satisfaction globale et varient entre 1 et 5. Chaque utilisateur évalue au moins 20 films. Nous normalisons les évaluations des films pour qu'ils correspondent à des valeurs comprises entre 0 et 1.

WS-Dream [2] Nous avons également choisi de tester notre approche sur le jeu de données WS-Dream qui fournit des résultats réels de la qualité de service QoS de 5825 services Web données par 339 utilisateurs. Les qualités de services évaluées sont le temps de réponse TD et le débit D. Afin de déduire l'évaluation de l'utilisateur nous calculons le ratio de la valeur obtenue de la qualité de service et la valeur souhaitée de TR et D. Ensuite nous agrégeons les deux ratios pour obtenir une valeur représentant l'opinion de l'utilisateur. Le jeu de données fournit également des informations sur les utilisateurs et les services. Nous considérons que le service Web est la ressource sur le Web que nous souhaitons évaluer.

Epinions [3] Epinions.com est un site réputé de partage d'opinions en ligne sur des objets tels que des produits électroniques ou cosmétiques, des entreprises, des films etc. Les objets sont évalués par des utilisateurs inscrits gratuitement et ayant des relations de confiance entre eux. Il existe peu de jeux de données de cette nature disponibles, la plupart des réseaux sociaux ne contiennent pas d'évaluations et les jeux de données tel que MovieLens et WS-Dream n'incluent aucun réseau social. La spécificité du jeu de données Epinions est la provision des deux. Dans nos expérimentations, nous utilisons un jeu de données contenant 664824 évaluations de 49290 utilisateurs sur 139738 objets. Nous considérons un objet comme une ressource.

6.3.2 Paramétrage

Dans le cadre de la mise en oeuvre de notre approche, plusieurs tests ont été effectués pour adopter le meilleur paramétrage. Les paramètres de l'algorithme flou C-means sont le nombre de clusters c et le critère de terminaison ϵ que nous fixons respectivement à 3 et 0.05. Le nombre de clusters a été fixé pour pouvoir représenter trois classes d'opinions sur les ressources sur le Web : Mauvais, Moyen et Bon.
Quant au critère de terminaison ϵ, nous l'avons choisi de sorte que nous obtenons des clusters bien définis mais en même temps chevauchés. Comme les degrés d'appartenances initiaux sont choisis au hasard, si le critère de terminaison est grand, l'algorithme s'arrête sur des valeurs aléatoires de degrés d'appartenance et donc avant que les clusters soient bien définis. Si le critère de terminaison est très petit le résultat se rapproche beaucoup à celui donné par un algorithme de classification classique tel que $\mathcal{K}$-means, nous obtenons alors des clusters bien séparés sans chevauchement ce qui nous permet pas de d'inclure les utilisateurs strictes dans l'opinion majoritaire.

Tableau 6.1 – Évaluation réelle versus fausse/stricte

Réelle	1	2	3	4	5
Fausse	5	5	1	1	1
Stricte	1	1	2	3	4

La première étape des expérimentations consiste à altérer au fil du temps un ratio des évaluations existantes dans le jeu de données pour certains utilisateurs et à préserver les évaluations pour le reste des utilisateurs. Ces utilisateurs deviennent malicieux ou bien stricts. Le tableau 6.1 montre un exemple de mapping entre les catégories des évaluations des utilisateurs honnêtes et modérés ainsi que ceux malicieux *versus* stricts. Les utilisateurs malicieux disent le contraire de ce qu'ils perçoivent, leur satisfaction est in-

versée. Les utilisateurs stricts exigent plus des ressources. Leurs évaluations sont alors plus restreintes/faibles que les utilisateurs normaux. A travers cette expérimentation, nous perturbons le système afin d'analyser son comportement en présence d'utilisateurs ayant des évaluations altérées.

6.3.3 Métriques

Dans nos expérimentations, nous utilisons trois métriques. La première est l'erreur moyenne s (*en anglais*, Root-Mean-Square Error (RMSE)). Cette métrique nous permet de comparer les valeurs de confiance obtenues avant et après avoir altéré les évaluations des utilisateurs. Les deux autres métriques sont la précision et le rappel. La précision mesure la capacité de notre modèle de crédibilité à rejeter les évaluations des utilisateurs malicieux. Le rappel mesure la capacité de notre modèle à trouver les évaluations des utilisateurs non malicieux. Les deux métriques sont exprimées en pourcentage.

6.3.4 Résultats expérimentaux

Pour chaque modèle proposé dans la thèse, nous avons opté pour des choix expérimentaux différents. Les choix porte sur le jeu de données choisi, les métriques ainsi que les paramètres d'évaluations.

6.3.4.1 Expérimentations sur le modèle de crédibilité flou

Nous avons réalisé trois expérimentations pour évaluer l'impact du modèle de crédibilité sur l'amélioration de la performance et la robustesse du système de confiance en utilisant le jeu de données MovieLens.

(a) **Expérimentation 1 :** Le calcul de la confiance se base sur deux paramètres : (i) le ratio des utilisateurs modifiés dans

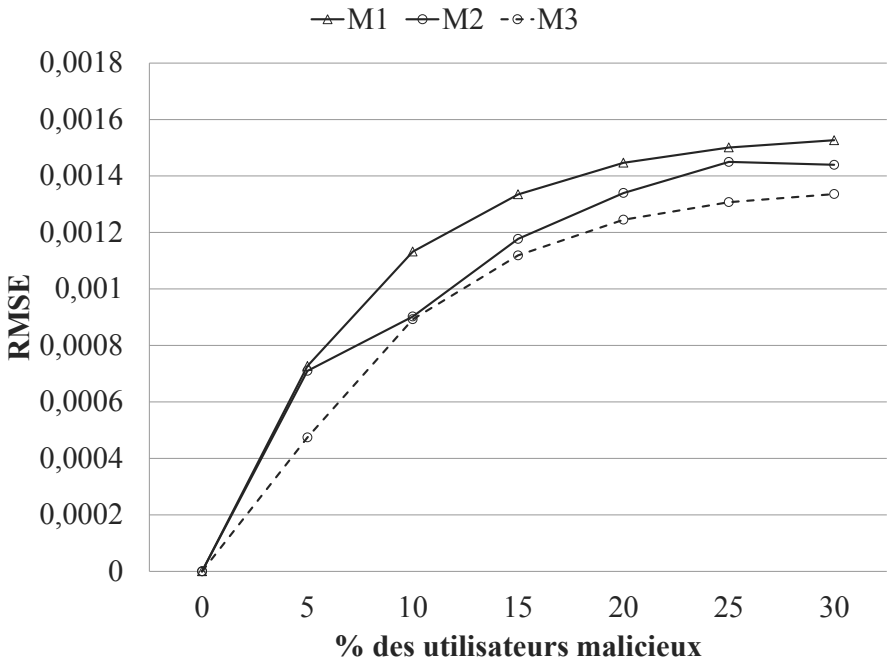
le jeu de données ; et (ii) le choix d'un modèle de confiance prédéfini $\mathcal{M}_i$. Nous considérons trois modèles de confiance : $\mathcal{M}_1$ utilise des évaluations non pondérées (sans considérer les valeurs de crédibilité des utilisateurs), $\mathcal{M}_2$ utilise le modèle de crédibilité basé sur le clustering $\mathcal{K}$-means, et $\mathcal{M}_3$ utilise un modèle de crédibilité basé sur le clustering C-means. Ensuite, nous comparons la valeur de confiance calculée avant et après l'introduction des utilisateurs malicieux en utilisant la métrique RMSE.

La figure 6.2(a) montre que l'erreur donnée par $\mathcal{M}_3$ est plus faible que celle donnée par le modèle $\mathcal{M}_2$. Ce qui signifie que notre modèle fournit des résultats plus proches de la valeur réelle de la confiance. Nous concluons alors que $\mathcal{M}_3$ contribue à l'amélioration de la robustesse du système de confiance.

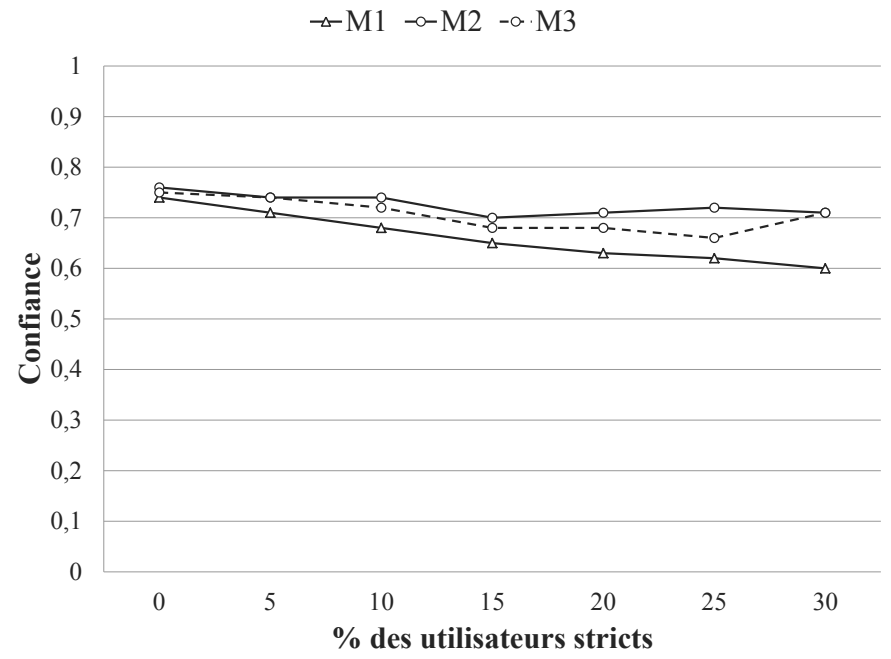
(b) **Expérimentation 2 :** Il s'agit de comparer les valeurs de confiance obtenues avec $\mathcal{M}_2$ et $\mathcal{M}_3$ en présence d'utilisateurs stricts. La figure 6.2(b) montre que les valeurs de la confiance obtenues avec $\mathcal{M}_3$ sont toujours inférieures à celles obtenues par $\mathcal{M}_2$. Les résultats obtenus montrent que les évaluations des utilisateurs stricts sont bien prises en compte en utilisant $\mathcal{M}_3$.

(c) **Expérimentation 3 :** Il s'agit d'analyser l'impact des différentes stratégies sur la performance du système de confiance à extrapoler les valeurs réelles de la confiance. Cette expérimentation établit l'avis de la majorité en utilisant les différentes stratégies proposées, calcule les valeurs de crédibilité des utilisateurs et enfin compare les valeurs de confiance obtenues.

La figure 6.3 montre que les stratégies modérée et forte donnent des résultats plus exacts et consistants. Ces stratégies restent stables même en altérant les évaluations des utilisateurs. Les résultats de la stratégie faible sont très instables et s'écartent considérablement de la valeur réelle de la confiance pour un ratio d'utilisateurs altérés dépassant les 15%. La stratégie faible



(a) *Robustesse*



(b) *Performance*

FIGURE 6.2 – Qualité de la confiance

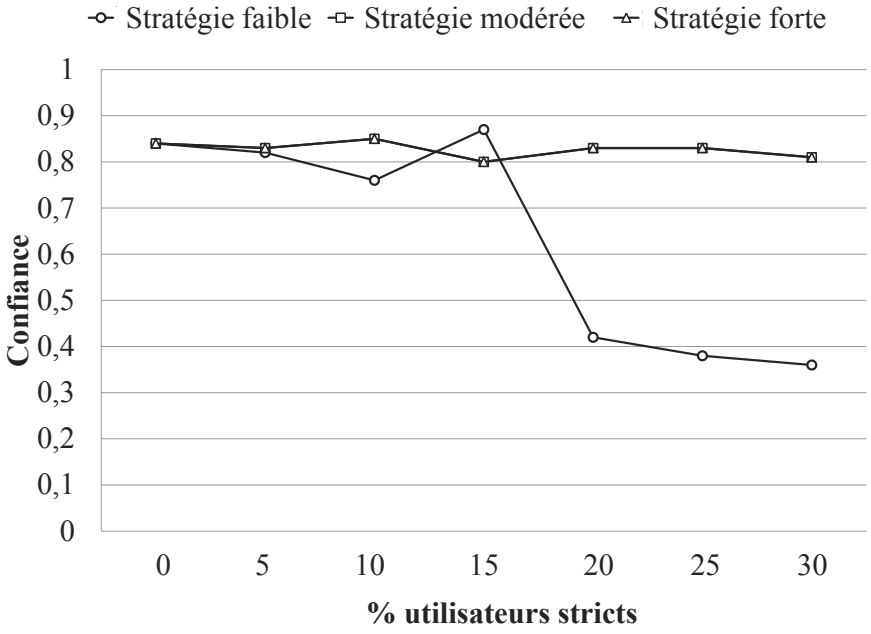
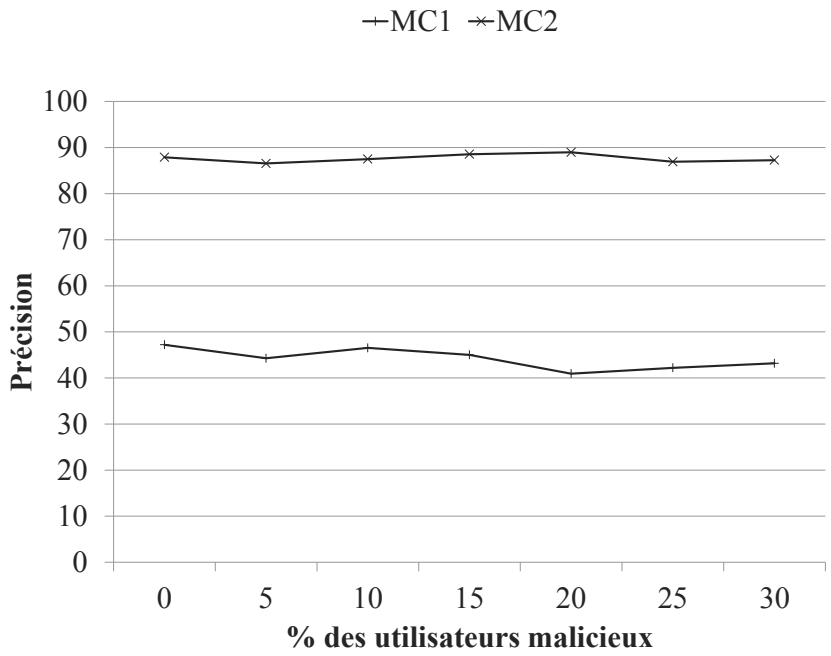


FIGURE 6.3 – Performance des stratégies

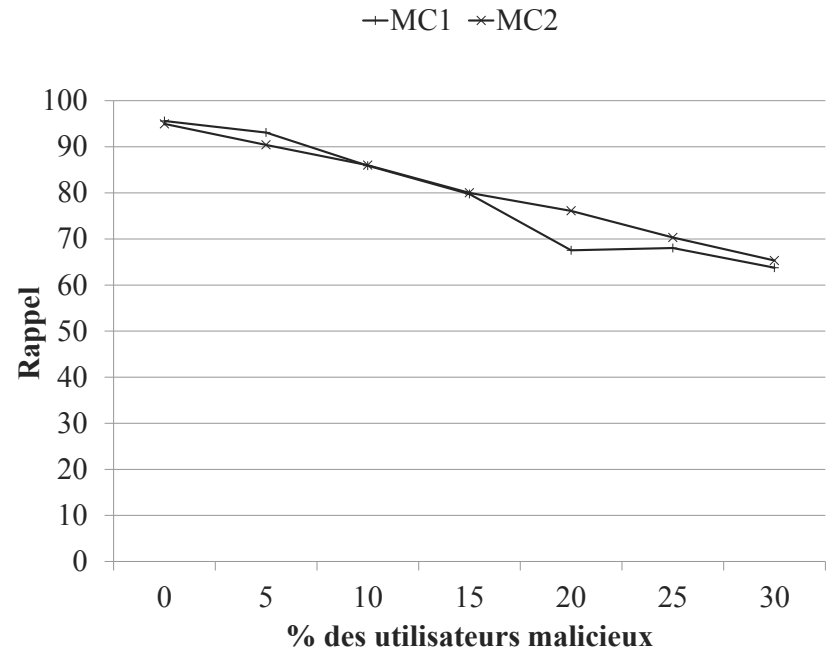
détecte un autre cluster qui pourrait potentiellement être le cluster majoritaire et le sélectionne pour être $\mathcal{M}_C$ s'avérant inadéquat.

(d) **Expérimentation 4 :** Il s'agit d'évaluer la capacité de notre modèle de crédibilité d'une part à rejeter les évaluations des utilisateurs malicieux et d'autre part à retrouver les évaluations des utilisateurs non malicieux. Nous utilisons pour cela les métriques de précision et rappel. Cette expérimentation a été réalisée sur le jeux de données WS-Dream pour la granularité de ses évaluations nous permettant de mieux évaluer les métriques de précision et rappel.

Nous considérons deux modèles de crédibilité : le premier basé sur le clustering $\mathcal{K}$-means ($\mathcal{MC}_1$) et le deuxième basé sur le clustering C-means ($\mathcal{MC}_2$). Les figures 6.4(a) et 6.4(b) montrent l'amélioration en précision et rappel atteinte par notre modèle de crédibilité. Malgré un faible rappel donné $\mathcal{MC}_2$



(a) *Précision*



(b) *Rappel*

FIGURE 6.4 – Précision & Rappel

comparé à celui donné par $\mathcal{MC}_1$, la précision reste inchangée. Les résultats montrent une augmentation moyenne 43% pour la précision et de 1.3% pour le rappel. L'augmentation de la précision est assez impressionnante et atteint 100% sur certains tests. Ce résultat montre l'efficacité de $\mathcal{MC}_2$ à rejeter les évaluations des utilisateurs malicieux.

6.3.4.2 Expérimentations sur le modèle de filtrage des évaluations

Nous avons effectué plusieurs expérimentations pour calculer des valeurs de confiance en fonction de deux paramètres : (*i*) le ratio des utilisateurs Sybil dans le jeu de données ; et (*ii*) le choix d'un modèle de confiance $\mathcal{M}_i$. Les expérimentations utilisent le jeu de données Epinions vu qu'il fournit en plus des évaluations, le réseau social des utilisateurs.

Nous proposons trois modèles pour évaluer la confiance. Les modèles diffèrent selon l'utilisation ou non d'un modèle de filtrage des évaluations et de sélection d'utilisateurs. Le premier modèle $\mathcal{M}_1$ calcule la confiance en utilisant l'ensemble des utilisateurs sans filtrage. Les deux autres modèles $\mathcal{M}_2$ et $\mathcal{M}_3$ utilisent un modèle de filtrage. $\mathcal{M}_2$ sélectionne les utilisateurs par une recherche en profondeur et $\mathcal{M}_3$ utilise la recherche heuristique basée sur la crédibilité proposé dans le chapitre 4.

Les trois modèles calculent la confiance selon l'approche probabiliste présentée dans le chapitre 5. Les évaluations des utilisateurs sont stockées dans une base de données probabiliste $\mathcal{BDP}rob$ ayant pour schéma $\mathcal{SP}rob(ressource, utilisateur, évaluation, p)$ où *ressource*, *utilisateur* et *évaluation* désignent respectivement l'identifiant du ressource, l'identité de l'utilisateur et le degré de satisfaction de l'utilisateur de cette ressource. Quant à l'attribut p, il correspond à la probabilité de l'existence d'un tuple dans la base de données ce qui correspond dans notre cas à l'incertitude sur la valeur d'une évaluation fournie par un utilisateur sur une ressource donné que nous associons à sa crédibilité de l'utilisateur.

Pour établir la confiance d'une ressource sur le Web à partir de $\mathcal{BDP}rob$, nous utilisons la requête SQL $SELECT\ AVG$ pour obtenir l'agrégation des évaluations.

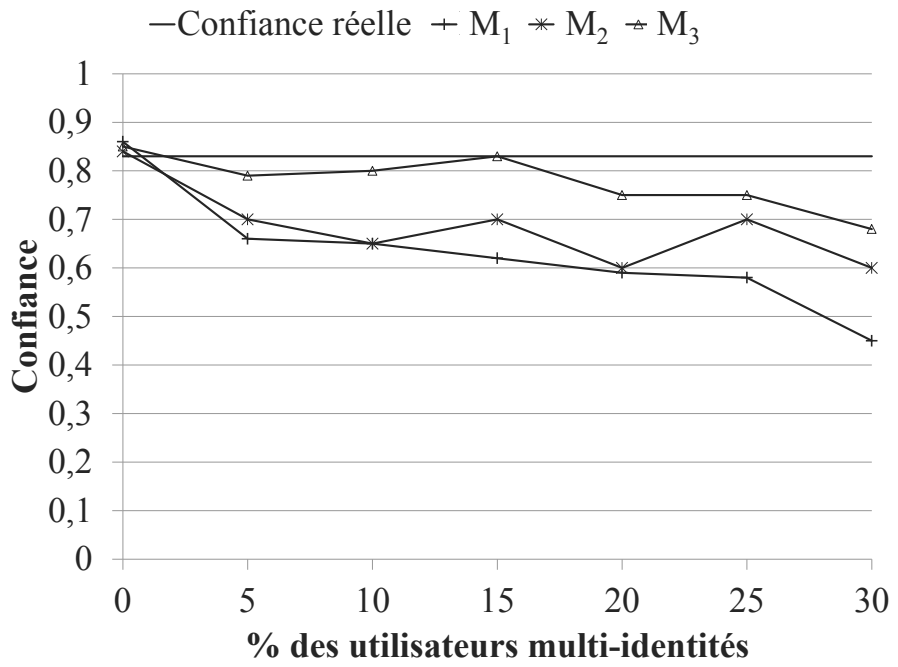


FIGURE 6.5 – Qualité de la confiance

La Figure 6.5 montre que les modèles $\mathcal{M}_2$ et $\mathcal{M}_3$ fournissent des valeurs de confiance plus exactes (i.e., plus proches de la valeur réelle de la confiance) que celles fournies par le modèle $\mathcal{M}1$. Cela montre l'importance de l'utilisation d'un mécanisme de filtrage des attaques Sybil et de sélection d'utilisateurs crédibles dans l'amélioration de la robustesse du modèle de confiance.

Cependant, nous constatons que les résultats de confiance donnés par $\mathcal{M}_2$ oscillent significativement comparés à ceux donnés par $\mathcal{M}_3$. Cela est dû au choix arbitraire des utilisateurs en utilisant la recherche en profondeur comparé à notre mécanisme de sélection basé sur les utilisateurs crédibles.

6.3.4.3 Expérimentations sur le modèle de confiance

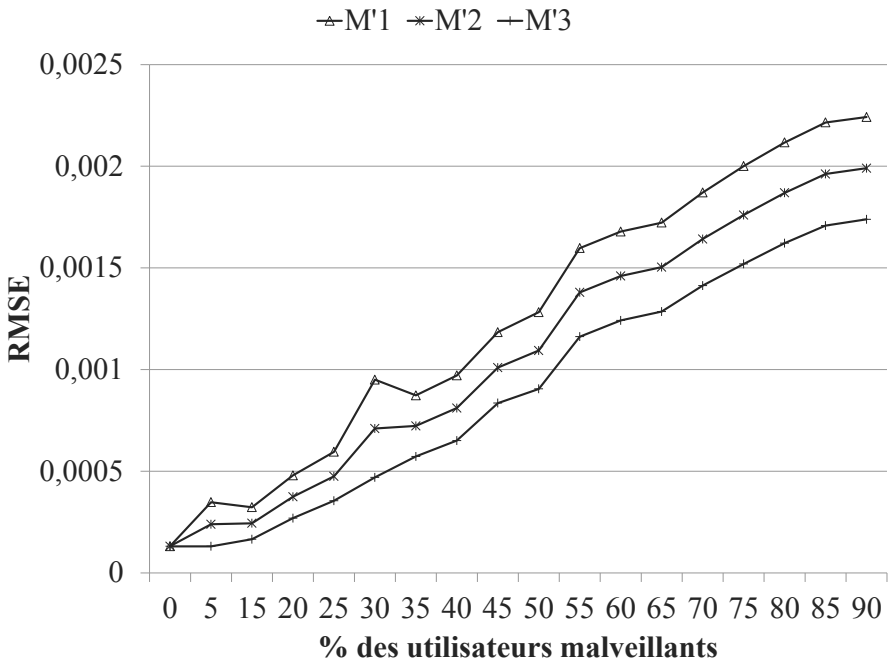


FIGURE 6.6 – *Qualité de la confiance*

La figure 6.6 analyse la performance de notre approche probabiliste de confiance définie dans le chapitre 5 en comparaison avec deux approches déterministes. Il s'agit de faire varier le ratio des utilisateurs malveillants et de calculer l'erreur moyenne (RMSE). Les approches reposent sur trois modèles de confiance différents M'_1, M'_2 et M'_3. M'_1 correspond au système Rateweb proposé dans [26]. M'_2 correspond à notre modèle déterministe de confiance proposé dans le chapitre 3 et consistant à agréger l'ensemble des évaluations avec comme poids la crédibilité des utilisateurs. M'_3 correspond à notre modèle probabiliste de confiance.

Nous constatons que l'erreur relative aux valeurs de la confiance obtenues par M'_1 et M'_2 oscille significativement comparée à celle obtenue par M'_3. Néanmoins, cette erreur reste supérieure à celle obtenue pour M'_3. A partir d'un ratio d'utilisateurs malveillants égal à 50 % l'erreur fournie par les différentes approches augmente

significativement.

6.4 Conclusion

Dans cette section, nous avons proposé WRTrust un système d'évaluation de la confiance des ressources sur le Web. Nous avons implémenté ce système à travers une application Web offrant des interfaces intuitives permettant de tester notre approche.
Nous avons également mené plusieurs expérimentations afin d'évaluation la performance et la robustesse de WRTrust. L'étude de la performance a montré une amélioration significative de la qualité de la valeur de confiance. L'étude de la robustesse a montré que notre système est moins sensible que les approches existantes lors de la présence d'utilisateurs malicieux, stricts et/ou Sybil.
Dans le futur, nous envisageons de mettre en place de nouvelles expérimentations, complémentaires de celles d'ores et déjà établies, en simulant d'autres types d'attaques.

Chapitre 7

Conclusion

Sommaire

7.1 Introduction

Cette thèse s'inscrit dans le domaine de la gestion de la confiance. Dans ce cadre, nous proposons une approche générique permettant l'évaluation de la confiance de tout type de ressource disponible sur le Web (ex. film, image, vidéo, service Web etc.). En effet, lorsqu'un utilisateur souhaite découvrir une ressource, il est confronté à un problème d'abondance des informations sur cette dernière. Notre approche a pour but de renforcer la confiance de l'utilisateur dans les ressources sur le Web et le guider dans ses choix. Vu l'importance que prête les utilisateurs actuels aux avis des autres utilisateurs, nous avons opté pour l'approche qui se base sur les retours d'expériences des utilisateurs sur l'utilisation des ressources en question. Ces avis sont présentés sous forme d'évaluations collectés sur le Web (ex. sites commerciaux, réseaux sociaux etc.).
L'étude de l'état de l'art des systèmes de confiance, présentée dans le chapitre 1, nous a permis de mettre l'accent sur deux principaux critères de succès d'un système de confiance : la robustesse contre les attaques des utilisateurs malveillants et la performance relatif à la qualité de la valeur de confiance. Afin de répondre à ces critères nous avons proposés trois modèles différents qui implémentent notre approche d'évaluation de la confiance.
Pour terminer, nous réalisons dans ce chapitre le bilan des contributions de notre travail. Nous rappelons l'objectif de chaque contribution et la solution adoptée pour l'attendre ainsi que les limites de chacune. Nous citons ensuite les perspectives envisageables dans le domaine de la gestion de la confiance.

7.2 Bilan des contributions

Modèle de crédibilité flou résilient aux évaluations biaisées. Le but de ce modèle est de rendre notre approche d'évaluation de la confiance robuste contre les attaques des évaluations biaisées. Afin

de limiter l'impact de ces évaluations, ce modèle estime la crédibilité des utilisateurs. La crédibilité d'un utilisateur détermine à quel point son évaluation sera prise en compte lors du calcul de la confiance.

Le modèle se base principalement sur deux notions : le consensus majoritaire et les utilisateurs stricts. Notre objectif principal est de trouver un consensus entre les utilisateurs afin d'inclure les avis utilisateurs stricts souvent exclus par les systèmes existants d'évaluation de la confiance. Nous avons réussi à inclure les évaluations de ces derniers dans l'avis de la majorité en utilisant la technique clustering flou et en particulier l'algorithme fuzzy C-means. Cette technique nous a permis de traiter l'incertitude de l'appartenance d'un avis à différents clusters d'avis. Nous avons alors raisonné sur les degrés d'appartenance des évaluations aux clusters afin d'établir des stratégies pour la recherche de l'avis de la majorité.

La stratégie faible attribue les évaluations aux clusters selon leurs degrés d'appartenance. Le cluster le plus peuplé est désigné comme le cluster majoritaire. Son centroide correspond par conséquent à l'avis de la majorité. Cette stratégie a réussi à inclure les avis des utilisateurs stricts vu que le centroide du cluster majoritaire se décale vers l'avis de ces derniers. Cependant elle était sensible au problème de recouvrement des clusters lorsque le critère de terminaison du clustering ϵ est grand. La stratégie modérée a résolu ce problème à travers l'ajout d'un seuil pour l'appartenance d'un avis à un cluster. La stratégie forte résout ce problème en se basant uniquement sur degrés d'appartenances des évaluations pour trouver l'avis de la majorité sans devoir décider si un avis appartient ou pas au cluster majoritaire.

Modèle de filtrage des évaluations. Le but de ce modèle est de rendre notre approche robuste contre les attaques Sybil.

Le modèle réalise alors un filtrage des évaluations en se basant sur les notions de crédibilité et des réseaux sociaux. Le modèle établit

un graphe de confiance extrait sur la base de crédibilité des utilisateurs et de relations de confiance dans le réseau social. Nous appliquons à ce graphe de confiance un mécanisme de distribution de capacités représentant le pouvoir d'évaluation que nous accordons aux utilisateurs. Cette opération permet de préparer le graphe au processus de filtrage réalisé à travers l'algorithme de flot maximal.

L'algorithme opère en parant d'une utilisateur fiable choisi comme source et en recherchant à chaque itération un chemin qui mène vers des utilisateurs crédibles et non Sybil. Le filtrage des évaluations en se basant sur le réseau social des utilisateurs nous a permis de réduire considérablement la capacité d'attaque des utilisateurs Sybil. L'ajout de la crédibilité comme heuristique de filtrage a permis de réduire d'avantage cette capacité et d'avoir des résultats plus surs.

Modèle probabiliste d'évaluation de confiance. Le but de ce modèle est de proposer une solution pour l'évaluation de la confiance d'une ressource sur le Web afin de lever l'incertitude sur son aptitude à satisfaire les demandes utilisateur.

Notre modèle est basé sur les notions de bases de données probabilistes associées à la sémantique de la théorie des mondes possibles. En effet, les bases de données probabilistes offrent une meilleure représentation de l'incertitude sous-jacente à la crédibilité des utilisateurs. Les évaluations de ces utilisateurs sont structurées sous forme d'une base de données probabiliste en utilisant deux modèles d'incertitude. Le premier est basé sur les tuples indépendants et la crédibilité des utilisateurs est associée à la probabilité des tuples. Le deuxième est basé sur les blocs indépendants (blocs de tuples) permettant l'utilisateur de plusieurs modèles de crédibilité. Ainsi à chaque évaluation sont associés plusieurs valeurs de probabilités correspondant aux crédibilités calculées par les différents modèles.

Avec le premier modèle nous avons réussi à réaliser un calcul in-

certain de la valeur de la confiance sous forme d'évaluation de requêtes. Nous avons utiliser à cet effet un algorithme permettant le calcul d'une approximation du résultat donné par l'opérateur $\mathcal{AVG}$ appliqué au bases de données probabilistes. Néanmoins, nous n'avons pas aboutit le calcul de la confiance dans le cas de la modélisation des évaluations sous forme d'une base de données probabiliste blocs indépendants. Vu qu'il n'existe à ce jour un algorithme permettant un calcul équivalent dans le cas de la modélisation basée sur les blocs indépendants.

Implémentation de l'approche. Finalement afin de mettre en oeuvre notre approche, nous avons conçu et implémenté un système d'évaluation de la confiance. Nous avons opté pour une architecture distribuée du système, en hébergeant ses différents composants du coté client.

L'implémentation du framework est réalisée à travers un prototype en JAVA. Le prototype nous permet de mener plusieurs expérimentations afin d'évaluer les différents modèles proposés en termes de performance et de robustesse. Afin d'évaluer critère performance, nous avons utilisé des métriques telles que l'erreur moyenne de la valeur de confiance (RMSE), la précision et le rappel. Et afin d'évaluer le critère robustesse, nous avons simulé des comportements malveillants tels que l'émission d'évaluations biaisées ou la création de fausses identités. Afin nous mettre dans des conditions réelles, nous avons exploité des jeux de données existants comportant des évaluations de ressources sur le Web et des informations sur les relations de confiance entre les utilisateurs. Les différentes expérimentations ont montré une amélioration significative de la qualité de la valeur de confiance.

7.3 Perspectives de recherche

Bien que la gestion de la confiance ait suscité l'intérêt de beaucoup de chercheurs, il demeure toutefois un grand nombre d'enjeux à examiner. Nous citons par la suite quelques pistes de recherche dans le domaine de la gestion de la confiance :

Incertitude : L'évaluation de la confiance est controversée par l'incertitude.
Il s'agit de chercher de nouveaux types d'incertitude présentes lors de l'évaluation de la confiance. L'ignorance de ces incertitudes mènera à l'inexactitude des valeurs de confiance. Une première piste serait de considérer les informations sur les transactions avec les services Web.

La sensibilité temporelle : L'idée de cette perspective de recherche est de mettre à jour les valeurs de crédibilités avec l'historique des évaluations données par l'utilisateur.

Evaluation de la confiance : L'approche probabiliste proposée évalue la confiance comme étant une requête d'agrégation en utilisant l'opérateur AVG. Nous avons utilisé un algorithme existant permettant de donner le résultat de cet opérateur dans le cas des bases de données probabilistes tuples indépendants. Cependant il n'existe pas un algorithme permettant l'exécution de l'opérateur AVG dans le cas des bases de données blocs indépendants.

Publications

2016

Revues internationales avec comité de lecture

- Zohra Saoud, Noura Faci, Zakaria Maamar & Djamal Benslimane. A Fuzzy-based Credibility Model to Assess Web Services Trust under Uncertainty. *Journal of Systems and Software JSS*.

2015

Revues nationales avec comité de lecture

- Zohra Saoud, Noura Faci, Zakaria Maamar & Djamal Benslimane. Un modèle de crédibilité basé sur le clustering flou pour une évaluation probabiliste de la confiance des ressources sur le Web. *Revue des Sciences et Technologies de l'Information Série ISI : Ingénierie des Systèmes d'Information*.

Conférences

- Zohra Saoud, Noura Faci, Zakaria Maamar & Djamal Benslimane. Sybil Tolerance and Probabilistic Databases to Compute Web Services Trust. *19th East-European Conference on*

Advances in Databases and Information Systems ADBIS, 11 septembre 2015, Poitiers (France) .

- Zohra Saoud, Noura Faci, Zakaria Maamar & Djamal Benslimane. Calcul de la confiance des services web dans un contexte d'utilisateurs muti-identités. *Colloque francophone sur l'ingénierie des protocoles-Nouvelles Technologies de la Répartition CFIP-NOTERE*. 24 juillet 2015, Paris (France) .
- Zohra Saoud, Noura Faci, Zakaria Maamar & Djamal Benslimane. Web Services Trust Assessment based on Probabilistic Databases. *The International Conference on NETworked sYStems NETYS*, 15 mai 2015, Agadir (Maroc) .

2014

Conférences internationales avec comité de lecture

- Zohra Saoud, Noura Faci, Zakaria Maamar & Djamal Benslimane.A Fuzzy Clustering-Based Credibility Model for Trust Assessment in a Service-Oriented Architecture. *23rd IEEE International Conference on Enabling Technologies : Infrastructure for Collaborative Enterprises WETICE*, 25 juin 2014, Parme (Italie) .

Interfaces graphiques de l'application Web WRTrust

Lorsqu'un utilisateur accède, pour la première fois, à notre site Web, une page d'accueil s'ouvre (voir Figure 1). Cette dernière affiche une brève description de notre application Web ainsi que les différents modèles l'implémentant. Un bouton "détails" est mis à la disponibilité de l'utilisateur afin de le rediriger vers les détails de chaque modèle ainsi que la liste de publications relatives à ce dernier.

La page affiche également un formulaire de connexion au site Web, dans le coin droit de l'écran. Si l'utilisateur n'est pas encore inscris, il peut le faire à travers la page d'inscription (voir Figure 2). Une fois sur cette page, l'utilisateur peut créer son compte personnel en indiquant : son Email, son prénom, son nom, un mot de passe et son affiliation. Une fois connecté au site Web, l'utilisateur peut choisir d'évaluation une certaine ressource sur le Web. Différentes pages sont disponibles pour lui permettre de fournir ses exigences. Elles sont toutes disponible à travers la barre verticale positionnée à gauche de l'interface Web.

La première page (Figure 3) est celle du paramétrage du modèle de crédibilité (CF. Chapitre 3). Les paramètres demandés sont le : le nombre de clusters, la précision ainsi que la stratégie de choix du cluster majoritaire dans le cas du clustering flou. Pour enregistrer les données, il suffit d'appuyer sur le bouton valider.

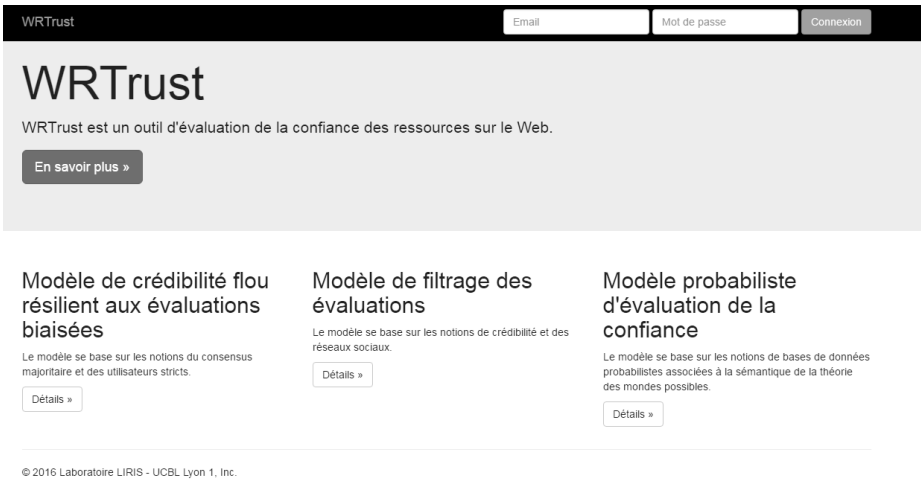


FIGURE 1 – Page d'acceuil

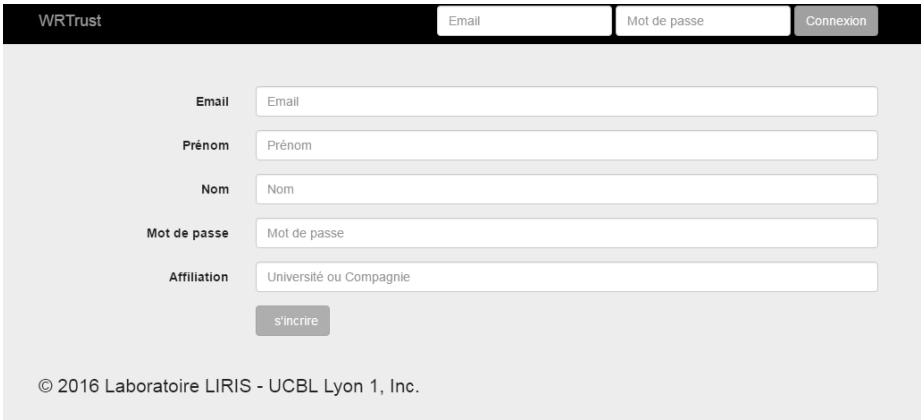


FIGURE 2 – Page d'inscription

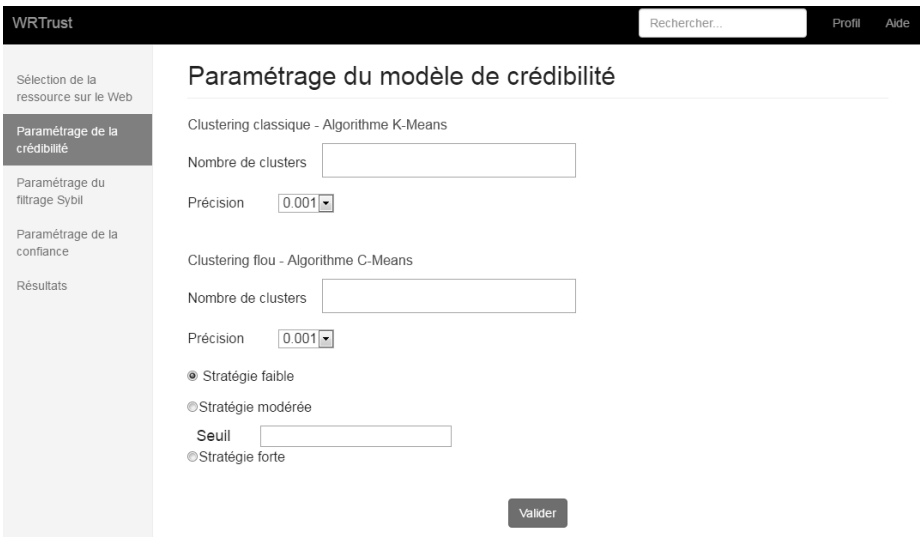


FIGURE 3 – Interface de paramétrage du modèle de crédibilité flou

La deuxième page (Figure 4) est celle du paramétrage du modèle de filtrage des évaluations (CF. Chapitre 4). Les paramètres demandés sont le : le seuil du nombre d'arcs entrants pour l'étape de l'élagage du réseau social, la capacité maximale pour l'étape de distribution des capacités et enfin l'heuristique pour l'étape de sélection des utilisateurs. De même, pour enregistrer les données, il suffit d'appuyer sur le bouton valider.

Le dernier paramétrage est celui de la confiance (voir Figure 5).

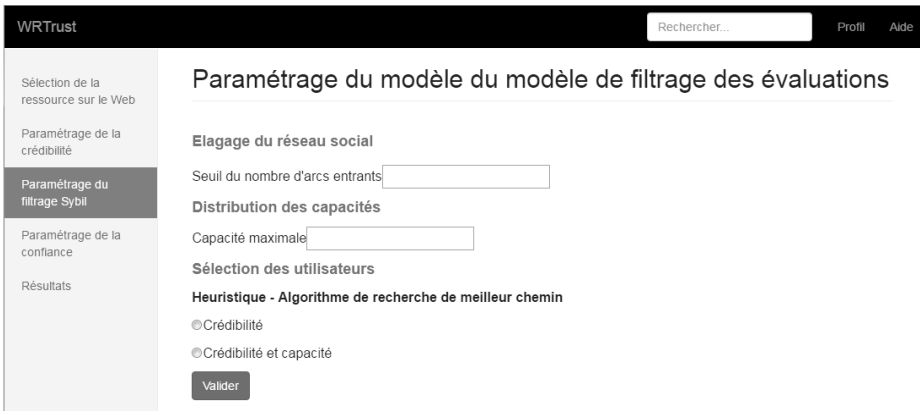


FIGURE 4 – Page de paramétrage du modèle de filtrage des évaluations

FIGURE 5 – Page de paramétrage de la confiance

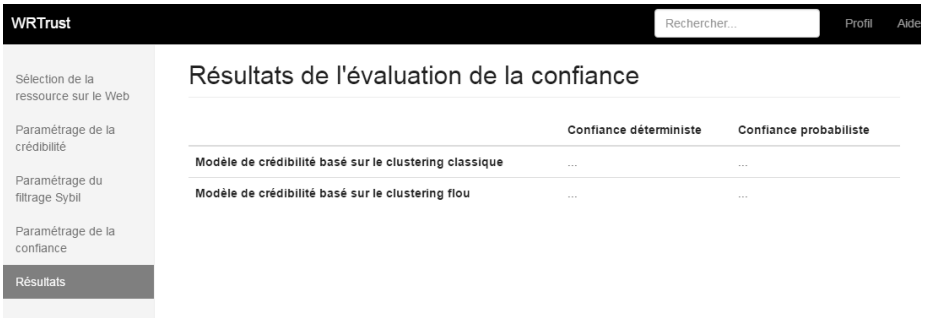


FIGURE 6 – Page des résultats d'évaluation de la confiance

Il suffit sur cette page de choisir l'approche de confiance à utiliser lors du calcul de la confiance : déterministe ou probabiliste. Si aucune approche n'est choisis, les résultats des deux approches sont affichés.
Ces résultats, l'utilisateur les retrouve dans page des résultats (voir figure 6).

Bibliographie

[1] J.C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, 1981.

[2] K.S. Bordens and I.A. Horowitz. *Social Psychology*. Psychology Press, 2001.

[3] R. Cavallo and M. Pittarelli. The theory of probabilistic databases. In *Very Large Data Bases Conferences*, Brighton, England, 1987.

[4] Alice Cheng and Eric Friedman. Sybilproof reputation mechanisms. In *Proceedings of the 2005 ACM SIGCOMM Workshop on Economics of Peer-to-peer Systems*, New York, NY, USA, 2005.

[5] F. Cornelli, E. Damiani, S. D. C. di Vimercati, S. Paraboschi, and P. Samarati. Choosing reputable servents in a p2p network. In *Proceedings of the 11th international conference on World Wide Web*, pages 376–386. ACM, 2002.

[6] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wisley, 1991.

[7] N. Dalvi and D. Suciu. Efficient query evaluation on probabilistic databases. *The VLDB Journal*, 16(4), 2007.

[8] G. Danezis and P. Mittal. Sybilinfer : Detecting sybil nodes using social networks. Technical report, 2009.

[9] L. R. Ford and D. R. Fulkerson. A simple algorithm for finding maximal network flows and an application to the hitchcock problem. *Canadian journal of Mathematics*, 1957.

[10] D. Gambetta. Can we trust trust ? In Diego Gambetta, editor, *Trust : Making and Breaking Cooperative Relations*, pages 213–237. Blackwell, 1988.

[11] T. Grandison and M. Sloman. Specifying and analysing trust for internet applications. In *Proceedings of the IFIP Conference on e-Commerce, e-Business and e-Government*, Lisbon, Portugal, 2002.

[12] R. Gupta and S. Sarawagi. Creating probabilistic databases from information extraction models. In *Proceedings of the international conference on Very large data bases*, volume 32, page 965. Citeseer, 2006.

[13] Patrick J Hayes and Harry Halpin. In defense of ambiguity. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 4(2) :1–18, 2008.

[14] C. Hota, M.S.V. Srikanth, A. Yla-Jaaski, J. Lindqvist, and K. Kristiina. Safeguarding against sybil attacks via social networks and multipath routing. In *Communications and Networking in China, 2007. CHINACOM '07. Second International Conference on*, 2007.

[15] J. Huang, L. Antova, C. Koch, and D. Olteanu. Maybms : a probabilistic database management system. In *SIGMOD Conference*, New York, USA, 2009.

[16] IEEE. Standard glossary of software engineering terminology. Technical report, IEEE Computer Society Press, 1990.

[17] R. Jampani, F. Xu, M. Wu, L. L. Perez, C. Jermaine, and P. J. Haas. Mcdb : a monte carlo approach to managing uncertain data. In *Proceedings of the 2008 ACM SIGMOD international*

conference on Management of data, pages 687–700. ACM, 2008.

[18] T. S. Jayram, S. Kale, and E. Vee. Efficient aggregation algorithms for probabilistic data. In *Annual ACM-SIAM Symposium on Discrete Algorithms*, New Orleans, USA, 2007.

[19] T. Kanungo, D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman, and A.Y. Wu. An efficient k-means clustering algorithm : Analysis and implementation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 24(7), 2002.

[20] A Kauffmann. Introduction à la théorie des sous-ensembles flous, à l'usage des ingénieurs. 1975.

[21] Y. Kim, M. T. Le, H. W. Lauw, E. P. Lim, H. Liu, and J. Srivastava. Building a web of trust without explicit trust ratings. In *Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference on*, pages 531–536. IEEE, 2008.

[22] W.A. Lesko. *Readings in Social Psychology : General, Classic and Contemporary Selections*. Boston : Allyn & Bacon, 1997.

[23] Y. Liu, A.H. Ngu, and L.Z. Zeng. Qos computation and policing in dynamic web service selection. In *Proceedings of the International World Wide Web Conference on Alternate Track Papers & Posters*, New York, USA, 2004.

[24] A. Lobna, F. Rim, and B. Djamal. Une approche de filtrage d'opinions à base de crédibilité dans un contexte de réseaux sociaux. *Ingénierie des Systèmes d'Information*, 20(4) :63–84, 2015.

[25] P.Tetali M. Mihail, A. Saberi. Random walks with lookahead in power law random graphs. *Internet Mathematics*, 1(1), 2007.

[26] Z. Malik and A. Bouguettaya. Rateweb : reputation assessment for trust establishment among web services. *Very Large Data Bases (VLDB) Journal*, 18(4), 2009.

[27] D Harrison McKnight and Norman L Chervany. The meanings of trust. 1996.

[28] A. Mislove, B. Viswanath, K.P. Gummadi, and P. Druschel. You are who you know : Inferring user profiles in online social networks. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM'10, 2010.

[29] T.H. Noor, Q.Z. Sheng, A.H.H. Ngu, A. Alfazi, and J. Law. Cloud armor : A platform for credibility-based trust management of cloud services. In *The ACM Conference on Information and Knowledge Management (CIKM)*, 2013.

[30] Judea Pearl. Heuristics : intelligent search strategies for computer problem solving. 1984.

[31] Josep M Pujol, Ramon Sanguesa, and Jordi Delgado. Extracting reputation in multi agent systems by means of social network topology. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems : part 1*, pages 467–474. ACM, 2002.

[32] S.D. Ramchurn, D. Huynh, and N.R. Jennings. Trust in multiagent systems. *Knowledge Engineering Review*, 19(1), 2004.

[33] T. Riggs and R. Wilensky. An algorithm for automated rating of reviewers. In *Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries*, pages 381–387. ACM, 2001.

[34] J. Sabater and C. Sierra. Reputation and social network analysis in multi-agent systems. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems : Part 1*, Bologna, Italy, 2002.

[35] D.A. Schum and J.R. Morris. Assessing the competence and credibility of human sources of intelligence evidence : contributions from law and probability. *Law, Probability and Risk*, 6(1), 2007.

[36] B. Schwartz. The paradox of choice : Why more is less. Ecco New York, 2004.

[37] G. Shafer. *A mathematical theory of evidence*, volume 1. Princeton university press Princeton, 1976.

[38] B. Sternthal, L.W. Phillips, and R. Dholakia. The persuasive effect of source credibility : A situational analysis. *The Public Opinion Quarterly*, 42(3), 1978.

[39] D. Suciu, D. Olteanu, Christopher R., and C. Koch. *Probabilistic Databases*. Morgan & Claypool Publishers, 2011.

[40] M. Tavakolifard and K. C. Almeroth. A taxonomy to express open challenges in trust and reputation systems. *Journal of Communications*, 7(7) :538–551, 2012.

[41] W. T. Teacy, J. Patel, N. R. Jennings, and M. Luck. Travos : Trust and reputation in the context of inaccurate information sources. *Autonomous Agents and Multi-Agent Systems*, 12(2), 2006.

[42] Nguyen Tran, Bonan Min, Jinyang Li, and Lakshminarayanan Subramanian. Sybil-resilient online content voting. In *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation*, Berkeley, CA, USA, 2009.

[43] M.C.M. Troffaes. Generalizing the conjunction rule for aggregating conflicting expert opinions. *International Journal of Intelligent Systems*, 21(3), 2006.

[44] Y. Wang and M.P. Singh. Formal trust model for multiagent systems. In *Proceedings of the International Joint Conference on Artifical Intelligence*, Hyderabad, India, 2007.

[45] J. Weng, C. Miao, and A. Goh. Protecting online rating systems from unfair ratings. *Trust, Privacy, and Security in Digital Business Lecture Notes in Computer Science*, 3592, 2005.

[46] A. Whitby, A. Josang, and J. Indulska. Filtering out unfair ratings in bayesian reputation systems. In *Workshop on Trust in Agent Societies hold in the Autonomous Agents and Multi Agent Systems Conference*, 2004.

[47] J. Widom. Trio : A system for data, uncertainty, and lineage. *Managing and Mining Uncertain Data*, 35, 2008.

[48] L. Xiong and L. Liu. Peertrust : Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering*, 16(7), 2004.

[49] R. R. Yager. Participatory learning : A paradigm for building better digital and human agents. *Law, Probability and Risk*, 3(1), 2004.

[50] B. Yu and M. P. Singh. An evidential model of distributed reputation management. In *International Joint Conference on Autonomous Agents and Multi-Agent Systems*, Bologna, Italy, 2002.

[51] Haifeng Yu, Phillip B. Gibbons, Michael Kaminsky, and Feng Xiao. Sybillimit : A near-optimal social network defense against sybil attacks. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, Washington, DC, USA, 2008.

[52] Haifeng Yu, Michael Kaminsky, Phillip B. Gibbons, and Abraham Flaxman. Sybilguard : Defending against sybil attacks via social networks. *SIGCOMM Comput. Commun. Rev.*, 2006.

[53] G. Zacharia, A. Moukas, and P. Maes. Collaborative reputation mechanisms for electronic marketplaces. *Decision Support Systems*, 29(4) :371–388, 2000.

[54] R. Zhou and K. 2007 Hwang. Powertrust : A robust and scalable reputation system for trusted peer-to-peer computing. *IEEE Transactions on Parallel and Distributed Systems*, 18(4), 2007.