



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE ABOU-BEKR BELKAID - TLEMCCEN

THÈSE

Présentée à :

FACULTE DES SCIENCES – DEPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

DOCTORAT EN SCIENCES

Spécialité: Informatique

Par :

Mme EL YEBDRI Zeyneb ép. BOUKLI-HACENE

Sur le thème

Un système de recommandation social et sémantique sensible au contexte

Soutenue publiquement le 18 Septembre 2021 à Tlemcen devant le jury composé de :

M ^r BENMAMMAR Badr	Professeur	Université de Tlemcen	Président
M ^r BENSLIMANE Sidi Mohammed	Professeur	ESI de Sidi Bel Abbès	Directeur de thèse
M ^{me} LAHFA Fedoua	Professeur	Université de Tlemcen	Co-Directrice de thèse
M ^r ADJOU DJ Réda	Professeur	Université de Sidi Bel Abbès	Examineur
M ^r AMAR BENSABER Djamel	Maître de Conférences A	ESI de Sidi Bel Abbès	Examineur
M ^r HADJILA Fethallah	Maître de Conférences B	Université de Tlemcen	Invité

*Laboratoire de Recherche en Informatique de Tlemcen (LRIT)
BP 119, 13000 Tlemcen - Algérie*

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA
RECHERCHE SCIENTIFIQUE
UNIVERSITÉ ABOU BEKR BELKAID TLEMCCEN
FACULTÉ DES SCIENCES
DÉPARTEMENT D'INFORMATIQUE

THÈSE

Présentée pour l'obtention du diplôme de
DOCTORAT
Spécialité : Informatique

Par

Mme EL YEBDRI Zeyneb ép. BOUKLI-HACENE

Un système de recommandation social et sémantique sensible au contexte

Thèse soutenue publiquement le 18 Septembre 2021 devant le jury :

Pr. BENMAMMAR Badr	Université de Tlemcen	Président
Pr. ADJOUDJ Réda	Université de Sidi Bel Abbes	Examineur
Dr. AMAR BENSABER Djamel	ESI de Sidi Bel Abbes	Examineur
Pr. BENSLIMANE Sidi Mohammed	ESI de Sidi Bel Abbès	Directeur
Pr. LAHFA Fedoua	Université de Tlemcen	Co-Directrice
Dr. HADJILA Fethallah	Université de Tlemcen	Invité

Remerciements

Je remercie en priorité **ALLAH LE TOUT PUISSANT** de m'avoir donné le courage, la force, et la volonté d'achever ce travail.

Tout d'abord, je tiens à remercier grandement mon Directeur de thèse Monsieur BENSLIMANE Sidi Mohammed, de m'avoir fait l'honneur d'accepter de diriger ma thèse, aussi pour sa disponibilité, ses précieux conseils et ses indications toujours fructueuses. J'ai beaucoup apprécié de travailler à ses côtés durant toutes ces années, j'avoue qu'il représente un encadreur exemplaire .

Je tiens à exprimer ma profonde gratitude à ma CoDirectrice de thèse, Madame LAHFA Fedoua, qui m'a donné la chance de réaliser ces travaux et de les mener à bien ainsi que pour la confiance et la liberté qu'elle m'a accordé durant toutes ces années.

J'adresse mes sincères remerciements à Monsieur BENMAMMAR Badr, Professeur à l'université de Tlemcen, qui m'a fait l'honneur de présider le jury de cette thèse.

J'exprime ma profonde reconnaissance à Monsieur ADJOU DJ Réda, Professeur à l'Université de Sidi Bel Abbes, Monsieur AMAR BENSABER Djamel Maître de conférences à ESI de Sidi Bel Abbes , et Monsieur HADJILA Fethallah Maître de conférences à l'université de Tlemcen, pour l'intérêt qu'ils ont bien voulu porter à ce travail en acceptant de l'examiner et d'en être rapporteurs.

Mes vifs remerciements vont à toutes les personnes ayant contribué de près ou de loin au bon déroulement et à l'aboutissement de cette thèse, tant sur le plan professionnel que sur le plan personnel.

En espérant n'oublier personne, j'adresse, enfin, un grand merci à mes collègues enseignants du département d'informatique.

À la mémoire de mon cher père qui me manque terriblement et
qui aurait été très fier de moi.

À ma chère mère en témoignage de ma profonde gratitude et de
mon incontestable reconnaissance, pour tous les sacrifices qu'elle
me contente et tout l'amour dont elle m'entoure.

À mon cher mari qui m'a toujours encouragé et soutenu.

À mes trésors : Meriem, Rayane et Imane

À mes chères Soeurs : Djaouida, Fatima Zohra et Wassila

À mes chers Frères que j'aime et qui m'adorent

À toute ma famille et amies ...

Résumé

Le travail présenté dans ce manuscrit se situe dans le domaine des systèmes de recommandation sensible au contexte (CARS) qui vise à améliorer les systèmes de recommandation traditionnels (SR) en prenant en compte les informations du contexte lors de la prédiction. Cependant, ces systèmes souffrent de certains défis, tels que le démarrage à froid et la rareté des données. De nouvelles méthodes sont proposées pour surmonter ces problèmes. Nous proposons dans notre travail, trois contributions. La première contribution consiste à palier les limites de l'approche du découpage sensible au contexte (Context-aware splitting approach (CASA)) qui représente l'une des approches de pré-filtrage les plus efficaces du système de recommandation sensible au contexte. Nous proposons d'ajouter principalement des informations de confiance ainsi sémantique pour améliorer la qualité de prédiction. La deuxième contribution consiste à proposer une approche hybride intitulée : Approche de post-filtrage sensible au contexte basée sur la confiance, qui utilise des déclarations de confiance comme une information riche avec la méthode de compensation de contexte qui appartient à l'approche de post-filtrage contextuel.

Les résultats des expérimentations révèlent que ces approches améliorent la pertinence des recommandations et surpasse les autres approches non contextuelles de la littérature en terme de précision.

Une autre approche que nous proposons : Approche de modélisation contextuelle basée sur la sémantique et la confiance (ST-CAMA), qui combine les informations de confiance et de contexte en utilisant la pondération contextuelle. Aussi, afin de sélectionner uniquement les voisins de confiance et qui ont des intérêts communs à l'item cible pour l'utilisateur actif, nous proposons de construire des clusters se basant sur des similarités sémantiques enrichis sémantiquement via les données ouvertes liées (LOD).

Mots clés : Système de recommandation, contexte, confiance, sémantique, méthode de compensation, système de recommandation sensible au contexte, système de recommandation basé confiance.

Abstract

The work presented in this manuscript is in the area of Context-Aware Recommender Systems (CARS) which aims to improve traditional recommendation systems (SR) by taking into account context information in predicting process. However, SR suffers from some challenges, such as cold start and data sparsity. New methods are proposed to overcome these problems. We propose in our work, three contributions. The first contribution aims to overcome the limitations of the Context-Aware Splitting Approach (CASA), which represents one of the most effective pre-filtering approaches of the context-aware recommender system. We propose to add mainly trust information as well as semantics to improve the quality of prediction. The second contribution aims in proposing a hybrid approach entitled: (Trust based Context aware Post Filtering Approach(TCPoFA)) , which belongs to contextual post-filtering approach. The results of the experiments reveal that these approaches improve the relevance of the recommendations and outperform other non-contextual approaches in the literature in terms of precision.

We propose another approach: Contextual modeling approach based on semantics and trust (ST-CAMA), which combines trust and context information using contextual weighting. Also, in order to select only trusted neighbors who have interests common to the target item for the active user, we propose to build clusters based on semantic similarities enriched semantically via linked open data (LOD).

keywords: Recommender system, context, trust, semantics, compensation method, context-aware recommender system, trust-aware recommender system.

Table des matières

Remerciements	i
Résumé	iii
Sommaire	v
Table des figures	ix
Liste des tableaux	xi
Liste des Algorithmes	xii
Liste des abréviations et des sigles	xii
1 Introduction générale	1
1.1 Contexte	1
1.2 Objectifs et problématiques	3
1.3 Contributions	4
1.4 Organisation du rapport	6
Partie I : Concepts de base et revue de littérature	9
2 Les systèmes de recommandation	10
2.1 Introduction	11
2.2 Historique	11
2.3 Définition des systèmes de recommandation (SR)	12
2.4 Concepts de base dans les SRs	13
2.4.1 Utilisateur	13
2.4.2 Item	14
2.4.3 Note (Vote ou évaluation)	14
2.5 Les approches des systèmes de recommandation	15
2.5.1 Les systèmes de recommandation basés contenu	16
2.5.2 Les systèmes de recommandation basés Filtrage collaboratif	17
2.5.2.1 Filtrage collaboratif basé sur la mémoire	17
2.5.2.2 Filtrage collaboratif basé sur le modèle	20
2.5.3 Les systèmes de recommandation hybrides	21
2.6 Méthodes d'évaluation des SRs	22

2.6.1	Métriques d'évaluation de la prédiction	23
2.6.2	Métriques d'évaluation des recommandations (Top-N)	24
2.6.3	Couverture	25
2.7	Limites des SRs	25
2.8	Conclusion	27
3	Les Systèmes de Recommandation sensibles au Contexte (CARS)	29
3.1	Introduction	30
3.2	Définition du contexte	30
3.3	Définition de sensibilité au contexte (context-awareness)	31
3.4	Type de contexte	32
3.4.1	Contexte de l'utilisateur	33
3.4.2	Contexte de l'item	33
3.4.3	Contexte de décision	33
3.5	Terminologies utilisées	33
3.5.1	Dimensions (ou facteurs) contextuelles	34
3.5.2	Conditions contextuelles	34
3.5.3	Situations contextuelles	34
3.6	Utilisation du contexte dans les systèmes de recommandation	34
3.7	Système de recommandation sensible au contexte	35
3.8	Formalisation du contexte	36
3.9	Les approches d'incorporation du contexte	37
3.9.1	Approche de Pré-filtrage Contextuel (contextual pre-filtering)	37
3.9.2	Approche de Post-filtrage Contextuel (contextual post-filtering)	38
3.9.3	Approche de la Modélisation Contextuelle (contextual modeling)	39
3.10	Bibliothèque CarsKit	39
3.11	Synthèse sur les CARS	40
3.12	Conclusion	42
4	Les systèmes de recommandation basés sur la confiance et la sémantique	43
4.1	Introduction	44
4.2	Web Social	44
4.3	Web sémantique	45
4.4	Confiance (Trust)	46
4.4.1	Web Social et confiance	46
4.4.2	Définitions	47
4.4.2.1	Confiance	47
4.4.2.2	Estimation de la confiance	47
4.4.2.3	Réseau de confiance	49
4.4.3	Propriétés de la confiance	50
4.4.4	Formalisation du réseau de confiance	52
4.5	Sémantique et données ouvertes liées	53
4.5.1	Mesures de similarité sémantique pour SR	54
4.5.1.1	Objectif des Similarités sémantiques	54
4.5.1.2	Approches de mesure de similarité sémantique	55
4.6	Systèmes de recommandation sensibles à la confiance	56
4.6.1	Architecture TARS avec confiance explicite et implicite	56
4.6.1.1	Architecture avec Confiance explicite	57

4.6.1.2	Architecture avec Confiance implicite	58
4.6.2	Métriques de confiance dans TARS	58
4.7	Systèmes de recommandation basé sur la sémantique	59
4.8	Conclusion	60
5	Revue de littérature sur les systèmes de recommandation sensibles au contexte basés sur la confiance et la sémantique	61
5.1	Introduction	62
5.2	Les systèmes de recommandation sensible au contexte	62
5.3	Les systèmes de recommandation basés sur la confiance (TARS) . . .	64
5.4	Les systèmes de recommandation sensibles au contexte basés sur la confiance	66
5.5	Les systèmes de recommandation basés sur la sémantique	67
5.6	Les systèmes de recommandation sensibles au contexte basés sur la sémantique	68
5.7	Synthèse et conclusion	69
	Partie II : Contributions	77
6	Approches proposées	78
6.1	Introduction	79
6.2	Approche de découpage du contexte basée sur le réseau de confiance (TBCASA)	80
6.2.1	Architecture de l'approche découpage du contexte basée sur le réseau de confiance (TBCASA)	80
6.2.2	Enrichissement des items et utilisateurs	82
6.2.3	Pré-filtrage des données contextuelles	82
6.2.4	Construction du réseau de confiance	84
6.2.5	Sélection des voisins de confiance	85
6.2.6	Filtrage collaboratif basé sur le découpage de contexte et la confiance	86
6.2.7	Recommandation	86
6.3	Approche de post-filtrage contextuelle utilisant un réseau de confiance	87
6.3.1	Architecture du système TCPoFA	88
6.3.2	Pré-compensation des données contextuelles(compensation des ratings)	88
6.3.3	Construction du réseau de confiance initial	90
6.3.4	Estimation des valeurs de notations inconnues	92
6.3.5	Degré de confidentialité des valeurs estimées	92
6.3.6	Reconstruction du réseau de confiance	93
6.3.7	Processus de prédiction	94
6.3.8	Post-filtrage contextuelle (compensation des ratings)	94
6.3.9	Algorithme de notre approche	94
6.4	Approche de modélisation contextuelle basée sur la sémantique et la confiance	94
6.4.1	Architecture de l'approche de modélisation contextuelle basée sur la sémantique et la confiance (ST-CAMA)	96
6.4.2	Création de cluster d'items	97
6.4.2.1	Enrichissement sémantique de description des items	97

6.4.2.2	Clustering des items	98
6.4.3	Sélection des voisins de confiance	98
6.4.3.1	Calcul de similarité des utilisateurs basé sur le contexte	98
6.4.3.2	Construction du réseau de confiance	101
6.4.4	Calcul de Prédiction des notes inconnues	102
6.5	Conclusion	102
7	Expérimentations	104
7.1	Introduction	104
7.2	Expérimentation du système TBCASA	105
7.2.1	Description de la base de Test	105
7.2.2	Résultats d'expérimentation	106
7.2.2.1	Paramètres de fractionnement	106
7.2.2.2	Résultats d'expérimentation	106
7.3	Expérimentation du système TCPoFA	108
7.3.1	Description de la base de Test	108
7.3.2	Résultats d'expérimentation	108
7.3.2.1	Impact de la propagation de la confiance	109
7.3.2.2	Impact de k (plus proche voisins de confiance) . . .	109
7.3.2.3	Impact du seuil des utilisateurs fiables	110
7.3.2.4	Impact des informations de confiance	110
7.3.2.5	Comparaison avec les approches de l'état de l'art . .	111
7.4	Conclusion	114
8	Conclusion et perspectives	115
8.1	Synthèse	115
8.2	Perspectives	116
	Liste des publications	118
	Bibliographie	119
	Bibliographie	130

Table des figures

2.1	Exemple de matrice de notation	13
2.2	Approches des Systèmes de recommandation	15
3.1	Matrice de notation avec données contextuelles	34
3.2	Utilisation du contexte dans les systèmes de recommandation	35
3.3	Les approches des systèmes de recommandation sensible au contexte	37
3.4	Intégration du contexte par rapport à chaque approche	38
3.5	Architecture bibliothèque carskit [Zheng et al., 2015a]	40
4.1	Évolution du Web	46
4.2	Zones de confiance [Haydar, 2014]	48
4.3	Propriété transitivité de la confiance	50
4.4	Propriété transitivité de la confiance	51
4.5	Exemple représentation d'un réseau de confiance	52
4.6	Exemple représentation de Matrice des données de confiance	53
4.7	Approche de mesure de similarité sémantique	55
4.8	Piliers du Système de recommandation basé sur la confiance	56
4.9	Architecture TARS basée sur la confiance explicite	57
4.10	Architecture TARS basée sur la confiance implicite	58
4.11	Les piliers des Systèmes de recommandation basé sur la sémantique	60
5.1	Synthèse des travaux connexes	74
5.2	État de l'art de l'approche de découpage contextuel	75
6.1	Architecture de l'approche découpage du contexte basé sur sémantique et contexte (TBCASA)	81
6.2	Présentation de l'approche de post-filtrage contextuelle basée sur la confiance (TCPoFA)	89
6.3	Compensation pré-contextuelle de la notation (pour une dimension contextuelle)	90
6.4	Architecture de l'approche de modélisation contextuelle basée sur la sémantique et la confiance (ST-CAMA)	97
7.1	Note moyenne dans différents contextes dans l'ensemble de données CiaoDVD	106
7.2	Note moyenne dans différents contextes	108
7.3	Impact de taux de propagation de confiance	109
7.4	Impact de k (plus proche voisins de confiance)	110
7.5	Impact du seuil des utilisateurs fiables Θ_c	111
7.6	Impact des informations de confiance	112

7.7 Comparaison des résultats avec différentes approches de recommandation 113

Liste des tableaux

2.1	Pourcentage de rareté dans quelques datasets	27
5.1	Synthèse d'état de l'art	70
6.1	Matrice de notation avant l'algorithme du découpage d'item	83
6.2	Matrice de notation après l'algorithme du découpage d'item	83
7.1	Principales approches de base utilisées dans notre expérimentation	107
7.2	Principales approches de base utilisées dans notre expérience	112

Liste des Algorithmes

- 1 Pseudo-code de l'approche post-filtrage contextuelles basée sur la confiance 95

Introduction générale

Sommaire

1.1	Contexte	1
1.2	Objectifs et problématiques	3
1.3	Contributions	4
1.4	Organisation du rapport	6

1.1 Contexte

De nos jours, les internautes passent beaucoup de temps sur divers sites Web et produisent de nombreuses informations (par exemple site de netflix¹). En parallèle, un grand nombre de plates-formes de commerce électronique (par exemple, Amazon²) et de réseaux sociaux en ligne (par exemple, Facebook³ et Twitter⁴) ont émergé qui génèrent également une énorme quantité de données. Avec une telle quantité d'informations, il est devenu difficile pour les utilisateurs de trouver facilement des produits répondant à leurs besoins et de prendre des décisions efficaces, ce que l'on appelle la surcharge d'informations. Un exemple de scénario typique de surcharge d'informations est décrit dans lequel l'utilisateur est exposé à un ensemble de films et ne sait pas lequel sélectionner. Si nous projetons cet exemple à des situations réelles où l'utilisateur est submergé par des milliers/millions d'articles, alors il est facile d'imaginer qu'il lui est très difficile de faire le bon choix sans aucune aide. Cependant, dans la pratique, les humains ne peuvent pas traiter autant d'informations sans l'aide d'un outil de filtrage automatique. Les systèmes de recommandation (SR) sont une

1. <http://www.netflix.com>
2. <https://www.amazon.co.uk/>
3. <https://www.facebook.com/>
4. <https://www.twitter.com/>

famille d'outils de filtrage d'informations qui se sont avérés être des moyens précieux pour aider les utilisateurs à trouver, de manière personnalisée, ce qui est pertinent pour eux dans des espaces d'informations complexes et débordants.

L'objectif principal des SR est d'aider les utilisateurs à satisfaire leurs besoins d'information lorsqu'ils traitent une quantité d'information importante. En d'autres termes, un SR est un système de filtrage qui tente de prédire la note ou les préférences qu'un utilisateur donnerait à un item (films, produits, hotels, etc.), qu'il n'a pas encore pris en compte.

Le filtrage basé sur le contenu (*Content-Based Filtering* (CBF)) et le filtrage collaboratif (*Collaborative Filtering*, CF) sont les deux approches de recommandation traditionnelles largement utilisées. Le filtrage basé sur le contenu, recommande généralement des items similaires aux items que l'utilisateur a apprécié dans le passé, et ce, en analysant la description de l'item et les préférences de l'utilisateur. Par exemple, si un utilisateur a aimé un film du genre comédie, le système lui recommandera très probablement un film du genre. Le filtrage collaboratif est considéré comme la technique de recommandation la plus populaire et efficace [Herlocker et al., 2002], qui prédit l'utilité des items pour un utilisateur particulier grâce aux items précédemment notés par d'autres utilisateurs similaires.

Toutefois, ces systèmes réalisent en général de bonnes performances en termes de précision de prédiction, mais dans certains domaines, les notes des utilisateurs ne précisent pas dans quelles conditions contextuelles l'item a été évalué. De plus, l'évaluation exacte d'un item peut être influencée lorsque ces conditions changent, d'où la nécessité de prendre en compte les informations de contexte. Les informations contextuelles décrivent la situation du comportement (exemple : le temps, la localisation ou la compagnie d'autres personnes). En prenant l'exemple du domaine de tourisme, un hotel bien noté en saison d'été, n'est pas forcément le cas si c'était en hiver (information contextuelle est : temps de visite).

Les systèmes de recommandation sensibles au contexte (*Context-Aware Recommender System* (CARS)) étendent le système de recommandation traditionnel en incorporant également le contexte, au plus des interactions utilisateur-item. Trois approches principales sont proposées pour intégrer les informations contextuelles dans la recommandation : *pré-filtrage contextuel*, *post-filtrage contextuel* et *modélisation contextuelle* [Adomavicius et al., 2005]. L'approche de pré-filtrage contextuel, les in-

formations contextuelles sont utilisées pour filtrer l'ensemble pertinent de données utilisateur en fonction d'un contexte cible. La recommandation est ensuite calculée en appliquant l'un des algorithmes de recommandation traditionnels sur les données sélectionnées. Dans l'approche de post-filtrage contextuel, l'algorithme de recommandation est appliqué sur les données d'entrée d'origine, puis le filtre est appliqué pour ajuster les recommandations générées en fonction du contexte cible. Pour l'approche de modélisation contextuelle, les informations contextuelles sont directement exploitées dans le modèle utilisé pour générer des recommandations. En raison de l'influence des informations contextuelles dans les systèmes de recommandation, une énorme quantité de travail a été mise en œuvre en utilisant l'une des trois approches, mais il y a tout de même une marge d'amélioration de ces systèmes.

1.2 Objectifs et problématiques

Nous nous intéressons dans notre travail à utiliser l'approche de filtrage collaborative dans les systèmes CARS. Certes, l'intégration de l'information contextuelle permet d'améliorer la qualité de la recommandation mais ne permet pas d'atténuer le problème de démarrage à froid et de rareté.

Dans cette thèse, nous visons à améliorer la qualité de la recommandation tout en prenant en considération d'autres informations pour surmonter les problèmes de démarrage à froid (cold start) et de rareté (sparsity) dans les systèmes de recommandation.

Dans la pratique, les gens font plus confiance aux opinions de leurs amis qu'aux étrangers. Cette relation de confiance affecte également leurs choix. C'était la motivation principale de plusieurs travaux de recherche (par exemple, [Sinha et al., 2001], [Jamali and Ester, 2009] et [Jamali and Ester, 2010]) qui ont tenté d'intégrer des informations de confiance dans le processus de prédiction pour faire des recommandations plus efficaces aux utilisateurs en fonction des évaluations d'amis de confiance dans le réseau de confiance.

En continuant avec l'exemple du domaine de tourisme pour le choix d'un hôtel, en plus de considérer la saison comme un contexte cible, l'utilisateur prend en compte les suggestions et recommandations de ses amis de confiance, et celles-ci compteront

plus que celles des étrangers. Les systèmes de recommandation basé sur la confiance (Trust-Aware Recommender System (TARS)) améliorent la prédiction des notes et résolvent les problèmes de filtrage collaboratif : démarrage à froid et rareté, car ils aident à connaître de nouvelles préférences des utilisateurs dans les systèmes de recommandation.

Aussi, les systèmes SR basés sur la sémantique ont marqué leurs succès dans différents domaines. De tels systèmes peuvent tirer parti du raisonnement sémantique pour fournir des recommandations beaucoup plus raisonnables dans le cas d'un nouveau item ou dans des ensembles de données très rares. L'utilisation des informations sémantiques décrivant les utilisateurs et les items peut améliorer la qualité des systèmes de recommandation. Ceci en fournissant des informations supplémentaires permettant d'améliorer dans la sélection des utilisateurs similaires en utilisant des similarités sémantiques.

1.3 Contributions

Les contributions de cette thèse sont :

1. Notre première contribution vise à améliorer l'approche de découpage sensible au contexte (Context-Aware Splitting approach(CASA)) [Zheng et al., 2014] qui appartient à l'approche de pré-filtrage contextuel. Les algorithmes de découpage de contexte sont considérés comme les plus efficaces et les plus populaires parmi les autres algorithmes de recommandation contextuelle [Baltrunas and Ricci, 2009], [Campos et al., 2013], [Baltrunas and Ricci, 2014] qui visent à produire une matrice de notation bidimensionnelle (2D). Trois types d'algorithmes ont été proposés dans CASA : découpage d'items (item splitting) [Baltrunas and Ricci, 2009], découpage d'utilisateurs (user splitting) [Said et al., 2011] et découpage item/utilisateur (UI splitting) [Zheng et al., 2014]. L'algorithme de découpage d'items considère qu'un item évalué dans deux contextes différents et noté différemment, est considéré comme deux items différents, mais à condition qu'il existe une différence significative (à l'aide des tests statistiques appelés : critères d'impureté [Baltrunas and Ricci, 2014]). Par conséquent, nous pouvons utiliser tous les algorithmes de recommandation traditionnels, qui fonctionnent sur une matrice de notation 2D. De même, pour l'algorithme de découpage des utilisateurs

teurs ; la différence consiste à diviser l'utilisateur en deux nouveaux utilisateurs virtuels au lieu d'item. Aussi, l'algorithme de fractionnement d'item/utilisateur (User-Item) fusionne les deux algorithmes précédents. Cependant, malgré le succès de CASA, cette approche souffre de problème de rareté des données après le fractionnement et démarrage à froid. Pour surmonter cette limitation, nous avons intégré la confiance qui permet d'un coté d'avoir des recommandations plus précises basées sur les évaluations d'amis de confiance afin de l'exploiter dans la phase de prédiction.

2. Nous proposons une approche de post-filtrage sensible au contexte basée sur la confiance (Trust based Context aware Post Filtering Approach (TCPoFA)) qui combine à la fois des informations de confiance et de contexte pour prédire des notes basées sur des évaluations des voisins de confiance et qui convient au contexte cible. Notre approche se compose de trois étapes principales : 1) Pré-traitement des données contextuelles en utilisant la méthode de compensation de contexte. 2) prédiction compensée basée sur le réseau de confiance. 3) Recommandation de filtrage post-contextuel.

Dans la première étape, nous appliquons la compensation de contexte pour calibrer les valeurs moyennes des différentes dimensions contextuelles. Dans la deuxième étape, nous construisons le réseau de confiance initial de l'utilisateur actif en utilisant une combinaison de valeurs de similarité et relations de confiance pour prédire la notation initiale. Ensuite, pour chacune de ces notations, une valeur de confidentialité est calculée qui sert à garder que les utilisateurs les plus importants dans le processus de recommandation finale. Enfin, la troisième étape consiste à calculer le score de notation par rapport à la situation contextuelle de l'utilisateur actif.

3. Une nouvelle approche est proposée pour surmonter les problèmes de SR et ce, en considérant les informations extraites de la connaissance du domaine sémantique des items. L'information sémantique permet de classer et de découvrir de nouvelles informations qui devraient fournir des recommandations plus efficaces. La majorité des travaux traitent du problème du démarrage à froid et rareté, mais négligent le problème des Multi-Intérêts et Multi-Contenu (MIMC). Ce problème se produit lorsqu'on considère les mêmes voisins pour

prédire tous les items invisibles pour l'utilisateur actif. Dans notre travail, nous proposons une approche hybride : Approche de modélisation contextuelle basée sur la sémantique et la confiance (ST-CAMA), qui combine les informations de confiance et de contexte en utilisant la pondération contextuelle. D'autre part, nous utilisons des informations sémantiques enrichies de LOD et en fonction de leur similarité sémantique (descriptions) pour classer les items (clustering), afin de sélectionner uniquement les voisins qui ont des intérêts communs à l'item cible pour l'utilisateur actif.

1.4 Organisation du rapport

Le reste de ce rapport de thèse est constitué de deux parties principales : la première est intitulée « Concepts de base et revue de littérature », elle introduit le contexte dans lequel se situe cette thèse et présente un état de l'art des travaux axés sur nos problématiques. La seconde partie est intitulée « Contributions ». Elle présente l'ensemble des propositions énoncées ci-dessus ainsi que l'ensemble des expérimentations effectuées.

1. La partie I : Concepts de base et revue de littérature. Cette partie inclut quatre chapitres :

- **Chapitre 2 : Systèmes de recommandation**

Ce chapitre présente une vue générale sur les systèmes de recommandation dits traditionnels ou classiques. Cette vue générale présente l'historique des systèmes de recommandation, suivie de leur définition, puis les différentes approches de recommandation qui existent, ainsi que leurs avantages et inconvénients.

- **Chapitre 3 : Systèmes de recommandation sensible au contexte**

Ce chapitre est consacré aux systèmes de recommandation sensible au contexte. Nous commençons par définir la notion de contexte. Le processus de prédiction contextuelle, nous enchaînons ensuite à définir les différentes approches d'intégration du contexte dans la recommandation. Nous terminons par présenter une bibliothèque intéressante CarsKit utilisée dans la partie expérimentation du domaine de CARS.

- **Chapitres 4 : Systèmes de recommandation basés confiance et sémantique**

Ce chapitre est consacré à l'introduction de deux types de travaux dans les CARS pour résoudre ces problèmes tout en maintenant la qualité des recommandations ; Principalement les SR basés confiance (TARS) puis ceux basés sémantique (SARS) et leurs impacts sur les CARS

- **Chapitres 5 : Revue de littérature sur les systèmes de recommandation sensibles au contexte basés sur la confiance et la sémantique**

Dans ce chapitre, nous avons élaboré un état de l'art sur les différentes approches proposées dans la littérature qui ont incorporé la confiance dans le processus de recommandation, de plus quelques travaux intégrant les techniques du web sémantique plus précisément les techniques de données ouvertes liées. Nous avons terminé ce chapitre par une étude comparative afin de mieux positionner nos contributions.

2. La partie II : Contributions. Cette partie regroupe deux chapitres :

- **Chapitre 6 : Approches proposées**

Ce chapitre présente nos trois contributions . En premier lieu, l'approche de découpage contextuelle basée sur la confiance (Trust-based Context-Aware Splitting Approach (TBCASA)) qui consiste à améliorer l'approche du découpage sensible au contexte (Context-Aware Splitting Approach (CASA)), et ce, en exploitant les avantages qu'offre les systèmes basés sur la confiance d'un coté ; et d'un autre coté, l'inclusion des informations sémantiques, qui permettent d'enrichir la description des items en recueillant des données auprès des LODs. Nous allons présenter l'architecture globale de cette approche. Ensuite, les différentes phases de l'approche sont expliquées.

En deuxième lieu, l'approche de post-filtrage sensible au contexte basée sur la confiance. Cette approche se base sur la méthode de compensation de contexte qui appartient à l'approche de post-filtrage contextuel. De

même, nous allons présenter l'architecture globale de cette approche, puis, ses différentes phases.

En troisième lieu, l'approche de modélisation contextuelle basée sur la sémantique et la confiance (Semantic and Trust based Context-Aware Modeling Approach (ST-CAMA)), qui combine les informations de confiance et de contexte en utilisant la pondération contextuelle, ainsi pour pouvoir sélectionner uniquement les voisins qui ont des intérêts communs à l'item cible pour l'utilisateur actif, nous construisons des clusters d'items se basant sur des similarités sémantiques enrichi via les données ouvertes liées(LOD). Nous allons présenter l'architecture globale de cette approche, puis, ses différentes phases.

- **Chapitre 7 : Expérimentations**

Ce chapitre est consacré à l'évaluation de nos contributions pour analyser leurs efficacité. Les expérimentations sont effectuées sur un jeu de données réel. En conséquence, nous présentons les résultats et les comparons avec les méthodes de recommandation de l'état de l'art.

Nous terminons cette thèse avec un chapitre conclusion et perspectives.

Partie I :
Concepts de base et revue de
littérature

Les systèmes de recommandation

Sommaire

2.1	Introduction	11
2.2	Historique	11
2.3	Définition des systèmes de recommandation (SR)	12
2.4	Concepts de base dans les SRs	13
2.4.1	Utilisateur	13
2.4.2	Item	14
2.4.3	Note (Vote ou évaluation)	14
2.5	Les approches des systèmes de recommandation	15
2.5.1	Les systèmes de recommandation basés contenu	16
2.5.2	Les systèmes de recommandation basés Filtrage collaboratif	17
2.5.2.1	Filtrage collaboratif basé sur la mémoire	17
2.5.2.2	Filtrage collaboratif basé sur le modèle	20
2.5.3	Les systèmes de recommandation hybrides	21
2.6	Méthodes d'évaluation des SRs	22
2.6.1	Métriques d'évaluation de la prédiction	23
2.6.2	Métriques d'évaluation des recommandations (Top-N)	24
2.6.3	Couverture	25
2.7	Limites des SRs	25
2.8	Conclusion	27

2.1 Introduction

Internet permet de lier des millions de services en ligne offrant une énorme quantité d'informations, de produits et de services (items). Plus particulièrement, un seul site Web de commerce électronique peut offrir jusqu'à des millions d'articles de différentes catégories. Par conséquent, sans surprise, lorsqu'un utilisateur recherche, par exemple, un livre à lire ou une destination touristique à visiter, peut être submergé par la quantité d'options à considérer. En fait, les utilisateurs peuvent avoir du mal à écarter des informations non pertinentes et à trouver des produits adaptés à leurs besoins spécifiques. Des systèmes de recommandation (SR) ont été proposés pour résoudre ces problèmes. Les SR ont été appliqués avec succès dans de nombreux domaines, notamment les livres, les films, les services de voyage et bien d'autres, afin d'aider les utilisateurs en ligne à maîtriser la surcharge d'informations en fournissant des recommandations personnalisées.

Dans ce chapitre, nous présentons d'une façon détaillée le domaine des systèmes de recommandation, en mettant l'accent sur les concepts de bases et les approches utilisées, ainsi nous décrivons les différentes métriques d'évaluation de ces systèmes.

2.2 Historique

Les systèmes de recommandation sont apparus comme un domaine de recherche individuel au milieu des années 1990 et ont dérivé de différents autres domaines de recherche tels que les sciences cognitives, la théorie d'approximation, la recherche documentaire, la théorie de la prévoyance et ont également des liens avec la science de la gestion et le marketing, dans la modélisation des choix du consommateur. Ces origines viennent de la notion du filtrage d'information. Ces derniers ont vu le jour dans le domaine de la Recherche d'Information (RI) quand la tendance s'est tournée vers la personnalisation des résultats présentés à l'utilisateur suite à sa requête afin de diminuer la surcharge d'informations [Adomavicius and Tuzhilin, 2005].

Puis, l'intérêt progressif porté à cette notion de filtrage a donné naissance aux systèmes de recommandation comme des systèmes voisins, mais indépendants des systèmes de RI du fait que les SRs sont spécialisés beaucoup plus dans le traitement des profils à long terme (les requêtes dans la RI représentent des profils à court terme).

Jusqu'à l'année 1997, les auteurs ont utilisé le terme « Filtrage Collaboratif » au lieu de « Système de Recommandation ». Cette dernière appellation a été stabilisée en cette année par Resnick et Varian [Resnick and Varian, 1997] dans lequel ils soutiennent que le filtrage collaboratif n'est pas la seule approche pour faire de la recommandation. De nombreux sites Internet courants utilisent des systèmes de recommandation pour fonctionner, tels que YouTube, Yahoo, Trip Advisor et Google.

2.3 Définition des systèmes de recommandation (SR)

La popularité croissante des systèmes de recommandation dans de nombreux domaines a conduit à l'apparition d'un conflit dans sa définition. Plusieurs définitions sont alors utilisées pour définir un système de recommandation. Nous citons, celles vu sous l'angle du e-Commerce : « *Recommendation Systems are software agents that elicit the interests and preferences of individual consumers [...] and make recommendations accordingly. They have the potential to support and improve the quality of the decisions consumers make while searching for and selecting products online.* » [Xiao and Benbasat, 2007].

Littéralement [Resnick and Varian, 1997] les SRs sont définis comme suit : « *People provide recommendations as inputs, which the system then aggregates and directs to appropriate recipients* ». Comme nous pouvons le voir, l'approche de Resnick et Varian est plus proche d'un concept de recommandation réel. Bien que cela semble être une définition appropriée pour le système de recommandation précoce à la fin des années 90, les systèmes de recommandation se sont beaucoup développés depuis. De ce fait, [Burke, 2002] propose une définition plus précise, ajoutant les notions de l'individualisation et la personnalisation, à savoir : « *Any system that produces individualized recommendations as output or has the effect of guiding the user in a personalized way to interesting or useful objects in a large space of possible options* »

Quatre caractéristiques clés des SRs ont été identifiées par : [Meyer, 2012] , à savoir :

- Aide à décider : prédire une note pour un utilisateur à un item
- Aide à la comparaison : classez une liste d'items de manière personnalisée pour un utilisateur

	I1	I1	I1	I1
U1	5	4	?	5
U2	5	?	5	2
U3	?	4	5	?
U4	3	1	?	1

Figure 2.1: Exemple de matrice de notation

- Aide à la découverte : fournir à un utilisateur des items inconnus qui seront appréciés
- Aide à l'exploration : donne des items similaires à un item cible donné.

2.4 Concepts de base dans les SRs

À la base de chaque RS, il y a trois éléments essentiels principaux qui sont : les *utilisateurs*, les *items* et les *notes* (évaluation ou rating).

Habituellement, ces informations sont représentées toutes ensemble au moyen d'une matrice de notation. Une telle matrice se compose d'un tableau où chaque ligne représente un utilisateur, chaque colonne représente un item spécifique et chaque entrée représente l'évaluation donnée par l'utilisateur à l'item particulier. Généralement, cette matrice est très creuse dans la pratique car les utilisateurs n'évaluent qu'une petite partie des items. La figure 2.1 montre un exemple de matrice de notation, où les utilisateurs expriment leurs préférences pour les items en utilisant une échelle d'évaluation à cinq points. Les items avec un point d'interrogation (note inconnue) sont invisibles pour l'utilisateur correspondant.

2.4.1 Utilisateur

Les utilisateurs sont au cœur de tout système de recommandation. c'est une personne qui interagit avec un système de recommandation, fournissant au système des évaluations afin de recevoir des recommandations ou des prédictions. L'ensemble des utilisateurs dans le système est représenté par U , où un utilisateur donné $u \in U$. Les utilisateurs peuvent être représentés de différentes manières selon les techniques de recommandation utilisées pour calculer les recommandations. Afin de fournir des

recommandations personnalisées, le système doit modéliser et conserver des informations sur leurs préférences.

2.4.2 Item

Dans les systèmes de recommandation, un item est tout ce qui est recommandé par un système de recommandation. Ce terme peut faire référence non seulement aux livres et aux films, mais également aux restaurants, aux pages Web ; aussi un individu ou un ensemble d'individus suggérés à l'utilisateur dans les réseaux sociaux. L'ensemble des items disponibles dans le système est représenté par I , où $i \in I$.

Les items peuvent être caractérisés par leur complexité et leur valeur ou leur utilité. Des exemples d'items de faible complexité et de faible valeur sont : les actualités, les pages Web, les livres, les films. Alors que les exemples d'items plus complexes et de plus grande valeur peuvent aller des téléphones portables, ordinateurs portables aux services financiers et voyages.

Selon le système et la technique de recommandation, le contenu de l'item peut être plus ou moins structuré et complexe. Cela peut aller d'un simple identifiant numérique dans un système de filtrage collaboratif à un ensemble de mots-clés ou à un ensemble de paires de valeurs d'attributs dans un système basé sur le contenu jusqu'à une description basée sur une ontologie dans des systèmes utilisant une ontologie de domaine.

2.4.3 Note (Vote ou évaluation)

Une note peut être binaire (aimer Ne pas aimer), à valeur entière ou réelle dans une échelle (la plus utilisée est score de 1 à 5) ou même unaire (une seule valeur de notation positive, mais aucune valeur de notation négative ou ambivalente). c'est un type de notation qui n'est pas directe. Par exemple, lorsqu'un utilisateur achète un article, il peut ne pas l'évaluer, mais l'acheter est généralement une indication que l'utilisateur est intéressé par l'article (Amazon suppose que l'utilisateur a noté 5 s'il a acheté l'article). Pour les évaluations à valeur entière et réelle, des nombres faibles indiquent généralement une préférence négative (l'article était mauvais), les nombres du milieu indiquent une ambivalence (l'article n'était ni bon ni mauvais) et un nombre élevé indique une préférence positive (l'article était très bon). Une note

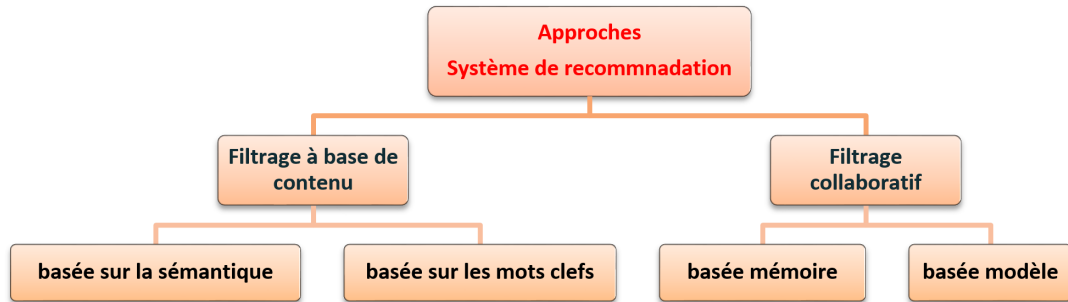


Figure 2.2: Approches des Systèmes de recommandation

donnée par un utilisateur u à un item i est représentée par r_{ui} .

Tout algorithme de filtrage collaboratif a besoin de cette matrice comme point de départ.

Une note peut être *explicite* attribuée directement par un utilisateur à un item en donnant une valeur numérique ou binaire à travers l'interface du système [Burke, 2002]. En outre, les préférences de l'utilisateur peuvent être déduites par le système en utilisant des algorithmes et techniques spécifiques, dans ce cas, on parle de note *implicite*.

2.5 Les approches des systèmes de recommandation

À travers les travaux de recherche, différentes tentatives de classification des approches ou des techniques ont été proposées. Parmi les approches les plus utilisées dans la littérature sont : le filtrage basé sur le contenu, le filtrage collaboratif et les approches hybrides selon [Adomavicius and Tuzhilin, 2005] (voir la Figure 2.2). Le filtrage basé contenu répond à : "Montrez moi plus que ce qui ressemble à ce que j'ai aimé", plutôt que le filtrage collaboratif répond à : "Dites moi ce qui est intéressant pour mes pairs", tant que les méthodes hybrides sont la combinaison des deux approches précédents.

2.5.1 Les systèmes de recommandation basés contenu

Les méthodes de recommandation basées sur le contenu (Content-Based Filtering CBF) sont largement inspirées du domaine de la recherche d'information. La différence réside dans l'absence de requêtes explicites formulées par l'utilisateur. Fondamentalement, les approches basées sur le contenu s'appuient sur le contenu des items pour les comparer à un profil lui-même constitué de thèmes. Chaque utilisateur du système possède alors un profil qui décrit ses propres centres d'intérêt. Le processus de recommandation consiste essentiellement à trouver de bonnes correspondances entre le profil utilisateur et les caractéristiques des items.

Cette approche présente des avantages tels que l'indépendance de l'utilisateur, puisque les systèmes CBF n'utilisent que les évaluations de l'utilisateur actif pour construire son modèle, par opposition aux techniques de filtrage collaboratif, qui reposent sur des «voisins». De plus, lorsqu'un nouvel item apparaît et n'a pas encore été évalué, les systèmes CBF peuvent le recommander. Il s'agit d'un problème de filtrage collaboratif bien connu, à savoir le problème de démarrage à froid . Cependant, ces techniques souffrent d'une sur-spécialisation, car elles ne sont pas capables de trouver des items inattendus : l'utilisateur recevra des recommandations d'items similaires à ceux qu'il les a noté auparavant. On distingue deux types de recommandation basé contenu, à savoir : basée sur les mots clefs et basé sur la sémantique.

1. basée sur les vecteurs de mots clefs

Dans les systèmes de recommandation basés sur le contenu s'appuyant sur le modèle d'espace vectoriel, les profils des utilisateurs et les items sont représentés comme des vecteurs de termes pondérés. La prédiction de l'intérêt d'un utilisateur sur un item donné peut être effectuée par calcul de similarité cosinus entre le vecteur de profil et le vecteur de l'item.

La plupart des systèmes de recommandation utilisent de simples modèles de recherche, comme la correspondance de mots-clefs ou le Modèle d'Espace Vectoriel (MEV) avec la pondération basique TF-IDF (Term Frequency-Inverse Document Frequency).

2. basé sur la sémantique

Les systèmes de recommandation basés sur la sémantique évoluent au rythme des méthodes et outils proposés dans le domaine du Web sémantique. Différentes stratégies ont été adoptées pour introduire de la sé-

mantique dans le processus de recommandation. La description de ces stratégies sont abordées en tenant compte de plusieurs critères :

- le type de source de connaissance impliquée (lexique, ontologie, etc.);
- les techniques adoptées pour l'annotation ou la représentation d'items ;
- le type de contenu inclus dans le profil utilisateur ;
- la stratégie de correspondance entre items et profil.

2.5.2 Les systèmes de recommandation basés Filtrage collaboratif

Le filtrage collaboratif (FC) se base sur le principe qui dit qu'on devrait pouvoir se servir de ce que les autres ont déjà trouvé et évalué pour trouver ce que nous cherchons. La notion de filtrage collaboratif est à la base de la recommandation, elle est basée sur le partage d'opinions entre les utilisateurs. Un système basé sur le filtrage collaboratif recommande à un utilisateur donné, les items hautement évalués par d'autres utilisateurs qui présentent des similarités dans leurs goûts et préférences. Autrement dit, l'idée clef est que la note de u pour un nouvel item i est susceptible d'être similaire à celle donnée par un autre utilisateur v , si u et v ont noté d'autres items d'une manière similaire. De même, u est susceptible de noter deux items i et j de la même façon, si d'autres utilisateurs ont donné des notes similaires à ces deux items. Supposons par exemple que les utilisateurs similaires de u ont apprécié l'hôtel x . u peut juger intéressant d'y aller. À la différence des approches basées sur le contenu, ces systèmes n'ont pas besoin d'information exogène sur les items et les utilisateurs. ([Aggarwal et al., 2016]).

D'après ([Bell and Koren, 2007], [Aciar et al., 2007], [Breese et al., 2013]), sur le plan algorithmique, il y a deux classes de filtrage collaboratif : les algorithmes basés sur la mémoire (*memory-based*) dits aussi basés sur les heuristiques (*heuristic-based*) et ceux basés sur les modèles (*model-based*).

2.5.2.1 Filtrage collaboratif basé sur la mémoire

On parle aussi d'approche des k plus proche voisins (ou k -Nearest Neighbours - k NN) ou dit heuristique. Cette approche automatise le principe de bouche-à-oreille,

où l'on se fonde sur l'avis de personnes partageant les mêmes idées ou d'autres sources fiables pour évaluer la valeur d'un item (film, livre, article, album, etc.) [Schafer et al., 2007]. Il considère la totalité des évaluations des usagers disponibles au moment du calcul de la recommandation. Les principaux avantages des recommandations basées sur la mémoire sont qu'elles sont simples à mettre en œuvre et que les recommandations générées peuvent être bien expliquées.

Ces approches sont regroupées en deux familles : **basés sur les utilisateurs** (*user-based collaborative filtering*) ou **basés sur les items** (*item-based collaborative filtering*).

- *basés sur les utilisateurs* : Les utilisateurs ayant des notes similaires sur certains items sont plus susceptibles d'avoir des notes similaires sur des items futurs.
- *basés sur les items* : Les items ayant des notes similaires dans le passé sont plus susceptibles d'avoir des notes similaires à l'avenir.

Le processus de calcul de la recommandation pour un utilisateur u passe par deux étapes successives :

1. **Phase du calcul du voisinage** En se basant sur le profil de l'utilisateur actif u , le système recherche les utilisateurs v (v diffère de u) qui lui sont les plus similaires. Deux mesures de similarité qui sont très utilisées sont : **la similarité vectorielle** et **la corrélation de Pearson**. Nous expliquons en détail les deux mesures dans ce qui suit.

- **La similarité vectorielle**

Dans cette méthode les utilisateurs u et v sont considérés comme deux vecteurs de même origine dans un espace de m dimensions, m étant le nombre d'items évalués par les deux utilisateurs (u et v). Plus deux utilisateurs sont similaires, plus l'angle entre leur vecteur est plus petit. Empiriquement, la similarité entre ces deux utilisateurs est calculée par la similarité Cosinus suivante (voir équation 2.1) :

$$sim_{u,v} = \frac{\sum_{i \in I_{u,v}} r_{u,i} \times r_{v,i}}{\sqrt{\sum_{i \in I_{u,v}} r_{u,i}^2} \times \sqrt{\sum_{i \in I_{u,v}} r_{v,i}^2}} \quad (2.1)$$

Avec

$I_{u,v}$: est l'ensemble des items notés en communs entre l'utilisateur u et v ,

$r_{u,i}$: est la note donnée par l'utilisateur u à l'item i .

Le problème avec la similitude cosinus est qu'elle ne tient pas compte des différences dans les notes moyennes des utilisateurs

- **la corrélation de Pearson**

La corrélation de Pearson telle utilisée par ([Resnick et al., 1994]; [Shardanand and Maes, 1995]) est une méthode issue des statistiques et fournit un ajustement de la similarité Cosinus. Elle est aussi **très utilisée** dans le domaine des systèmes de recommandation pour mesurer la similarité entre deux utilisateurs (des items cas basé sur items) et aussi la plus performante en terme de pertinence des recommandations [Schafer et al., 2007]. La formule ci-dessous 2.2, nous donne cette valeur pour deux utilisateurs u et v :

$$sim_{u,v} = \frac{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_{u,v}} (r_{v,i} - \bar{r}_v)^2}} \quad (2.2)$$

Avec

$I_{u,v}$: est l'ensemble des items notés en communs entre l'utilisateur u et v ,

\bar{r}_u (\bar{r}_v) : est la moyenne des notations données par l'utilisateur u (v),

$r_{u,i}$: est la note donnée par l'utilisateur u à l'item i .

De la même manière, en utilisant la corrélation de Pearson , nous pouvons calculer la similitude des items (cas proche voisin basé item) i et j qui ont été notés par les utilisateurs. Au lieu de se baser sur les $I_{u,v}$ qui sont les items notés en commun par deux utilisateurs, ça sera, $U_{i,j}$ qui représente les utilisateurs qui ont noté en commun les deux items.

2. **Phase de prédiction** Une fois que toutes les similarités de l'utilisateur cible par rapport aux autres utilisateurs sont calculées, les n utilisateurs les plus similaires qui constituent le voisinage de cet utilisateur cible sont définis, la prédiction de la valeur d'un item i évalué par l'utilisateur u , est calculée à l'aide de la formule suivante (voir équation 2.3) :

$$\hat{r}_{u,i} = \frac{\sum_{v \in N_u} sim_{u,v} r_{v,i}}{\sum_{v \in N_u} sim_{u,v}} \quad (2.3)$$

Avec

N_u : nombre d'utilisateurs présents dans le voisinage de u , ayant déjà voté sur l'item i

$sim_{u,v}$: valeur de mesure de similarité entre u et v

$r_{v,i}$: note de l'utilisateur v pour l'item i .

Par ailleurs, tous les utilisateurs sont différents dans leurs façons de noter un item. En effet, il existe des utilisateurs qui notent large en affectant la valeur de 5 sur une échelle de 1 à 5 pour un item qu'ils jugent satisfaisant alors que d'autres, qui ont tendance à noter de façon plus stricte, attribueront la valeur 3 à un item qu'ils jugent satisfaisant. Pour compenser la variation dans les jugements des utilisateurs, la note de chaque utilisateur est ajustée par la moyenne de ses notes. Cependant la formule finale pour le calcul de prédiction est la suivante (équation 2.4) :

$$\hat{r}_{u,i} = \bar{r}_u + \frac{\sum_{v \in N_u} sim_{u,v}(r_{v,i} - \bar{r}_v)}{\sum_{v \in N_u} sim_{u,v}} \quad (2.4)$$

Avec :

\bar{r}_u : représente la moyenne des notes de l'utilisateur u

Il a été démontré par la suite que la restriction aux k plus proches voisins améliore considérablement la qualité des recommandations fournies [Herlocker et al., 2017]. Une analyse des données est nécessaire pour fixer la valeur de k , cette valeur peut dépendre sensiblement du domaine et des données utilisées. Selon [Herlocker et al., 2002] $k = 20$ est généralement une bonne valeur et les valeurs entre 20 et 50 utilisateurs restent raisonnables.

2.5.2.2 Filtrage collaboratif basé sur le modèle

Les algorithmes basés sur le modèle se basent aussi sur les notes précédentes (les profils) des utilisateurs, sauf que cette fois-ci, ces notes sont utilisées pour construire un modèle prédictif par apprentissage. On ne calcule pas directement les prédictions, mais on essaie de classifier les utilisateurs suivant des groupes ou d'apprendre les modèles à partir de leurs données. Une fois les groupes ou les modèles d'utilisateurs sont trouvés, la prédiction pour un utilisateur donné est générée automatiquement à partir de son profil. Pour la construction du modèle, plusieurs méthodes sont uti-

lisées. Des méthodes statistiques pour construire les modèles, la première est très utilisée dans le domaine de l'apprentissage machine : Les modèles de classification (*Cluster models*). La deuxième méthode est celle des réseaux Bayésiens. En général, les méthodes basées sur le modèle utilisent les techniques d'apprentissage machine et les techniques statistiques pour apprendre le modèle à partir des profils des usagers.

Les systèmes de recommandation basés sur des modèles de facteurs latents offrent des capacités expressives hautes pour décrire les divers aspects des données. Ainsi, ils tendent à fournir des résultats plus précis que les systèmes basés sur le voisinage. Cependant, la grande partie de la littérature et des systèmes commerciaux (par exemple, Amazon , TiVo et Netflix (Netflix)) sont basés sur le voisinage [Picot-Clémente, 2011]. Actuellement, il existe beaucoup plus de systèmes de recommandation basés sur le voisinage, car ils sont considérés comme plus faciles et intuitifs à manipuler. Tout d'abord, ils fournissent naturellement des explications plus intuitives du raisonnement derrière les recommandations, ce qui améliore l'expérience utilisateur. Enfin, ils peuvent immédiatement délivrer des recommandations à l'utilisateur en se basant sur les feedbacks qu'il vient juste de fournir.

Dans le cadre des approches basées modèles, la prédiction peut être faite de deux façons différentes :

- A partir de la prédiction fournie par le modèle lui-même, en construisant par exemple un modèle probabiliste pour l'estimation des valeurs de prédiction ou directement à partir du modèle.
- Ou bien, en regroupant les utilisateurs/items par les méthodes de clustering et par la suite, les méthodes basées mémoires (basées utilisateurs ou basées items) seront utilisées pour prédire les évaluations pour les items.

2.5.3 Les systèmes de recommandation hybrides

L'approche hybride combine deux ou plusieurs autres approches natives. Chacune des deux principales approches de recommandation de cette section utilise une source d'information différente pour générer la recommandation. Une méthode pour améliorer la qualité de la recommandation consiste à utiliser une combinaison des approches. Dans cette sous section, nous introduisons des méthodes pour combiner les résultats d'autres systèmes afin d'améliorer la qualité de la recommandation ([Panniello et al.,

2009], [Burke, 2002]). Trois types d'hybridations sont citées dans la littérature ([Abasi et al., 2014], [Li, 2021]) :

- **Hybridation monolithique**

Dans cette approche, il n'y a qu'un seul système de recommandation. Ce système de recommandation utilise différentes sources et approches pour faire la recommandation. Ce cas est donc une modification du comportement de base des algorithmes de recommandation pour en faire un autre qui peut utiliser les avantages de tous les autres algorithmes .

- **Hybridation parallélisée**

Dans cette approche, nous utilisons plus d'un algorithme de recommandation. Le processus d'hybridation obtient les résultats des systèmes de recommandation, les combine et génère la recommandation finale.

Nous pouvons utiliser différentes techniques pour combiner les résultats d'autres systèmes de recommandation, tels que mixte (une union des résultats de tous les systèmes de recommandation), pondéré (une combinaison pondérée des résultats), commuté (utiliser les résultats de systèmes de recommandation spécifiques pour des tâches) et le vote à la majorité.

- **Hybridation de pipeline**

Dans cette approche, nous utilisons également plusieurs systèmes de recommandation et nous plaçons les systèmes de recommandation dans un pipeline ; Le résultat d'un recommandant est l'entrée d'un autre. Le premier recommandeur peut créer un modèle de l'entrée et le transmettre au prochain système de recommandation ou peut générer une liste de recommandations à utiliser par le prochain système de recommandation.

2.6 Méthodes d'évaluation des SRs

Afin d'évaluer la performance des systèmes de recommandation, un certain nombre de mesures statistiques ont été proposées dans la littérature [Herlocker et al., 2004]. On s'intéresse dans cette section aux mesures d'évaluation de la précision des prédictions et de la qualité des recommandations. L'évaluation de performance en SR dans la littérature est souvent limitée au calcul de la précision de prédiction [Herlocker

et al., 2004]. La précision mesure, en général, la différence entre les valeurs des notes prédites par le système de recommandation et les valeurs réellement fournies par les utilisateurs. L'évaluation de systèmes de recommandation suit, en général, une des trois méthodes : hors ligne, études sur un échantillon d'utilisateurs ou évaluation en ligne [Shani and Gunawardana, 2011].

Nous nous intéressons dans notre travail à l'évaluation hors ligne. cette dernière est la plus simple à réaliser et la moins risquée. Il s'agit globalement de diviser les données disponibles en deux parties, la partie d'apprentissage et la partie test, avant d'utiliser la partie d'apprentissage pour prédire la partie test. Elle permet d'intégrer facilement une grande masse d'utilisateurs. Elle n'est pas très sensible aux potentiels changements dans le comportement de l'utilisateur.

On distingue deux stratégies selon lesquelles les SR communiquent leurs résultats aux utilisateurs. Dans la première, le système répond à la question : est-ce que l'utilisateur va apprécier cet item ? Ces systèmes cherchent à prédire toutes les valeurs manquantes de la matrice de notes, et affichent leur valeur prédite à côté de l'item lors de sa consultation par l'utilisateur, Movielens est un exemple de cette stratégie. Dans la deuxième, le SR répond à la question suivante : Quels sont les items que l'utilisateur va apprécier ? Les systèmes qui suivent cette stratégie donnent en sortie une liste ordonnée des meilleurs items que l'utilisateur va apprécier, la valeur numérique de la note prédite n'est pas une priorité, l'essentiel est que la liste contienne des items pertinents. La mesure d'évaluation doit s'accorder avec la stratégie suivie par le système.

Nous présentons dans ce qui suit deux types de métriques, pour l'évaluation de la prédiction et des recommandations Top-N, respectivement [Herlocker et al., 2004].

2.6.1 Métriques d'évaluation de la prédiction

Pour mesurer la précision de la prédiction, différentes mesures d'évaluation de la prédiction ont été présentées afin de juger la performance des systèmes à base du FC. On compare la note réelle à la note prédite. La mesure la plus connue dans cette catégorie est l'erreur moyenne absolue (Mean Absolute Error (MAE)) (2.5). $\tilde{r}_{u,i}$ est la note prédite pour un couple (utilisateur, item), $r_{u,i}$ est la note existante pour

ce couple et N est le nombre total de notes. La Root Mean Square Error (RMSE) accentue les erreurs larges (voir équation 2.6). Quelle que soit la mesure, plus l'erreur est basse, plus la note prédite est proche de la note réelle et donc meilleure est la recommandation.

$$MAE = \frac{\sum_{u,i} |r_{u,i} - \tilde{r}_{u,i}|}{N} \quad (2.5)$$

$$RMSE = \sqrt{\frac{\sum_{u,i} |r_{u,i} - \tilde{r}_{u,i}|^2}{N}} \quad (2.6)$$

Avec :

$r_{u,i}$: est la note réelle

$\tilde{r}_{u,i}$ est la note prédite.

N Nombre total des prédictions.

2.6.2 Métriques d'évaluation des recommandations (Top-N)

Ces mesures ne prennent pas en compte la valeur que le SR prédit, mais elles considèrent sa décision de recommander ou non un item. Le SR est récompensé pour les bonnes décisions (l'intégration des items pertinents dans sa liste de recommandations), et pénalisé pour les mauvaises (l'intégration des items non pertinents dans la liste, ou l'absence d'un item pertinent de la liste). L'objectif est de mesurer la fréquence des bons et mauvais jugements portés par le système de recommandation à l'égard des items.

La *F-mesure* est la mesure la plus utilisée de cette catégorie [Herlocker et al., 2004]. Elle se compose de deux valeurs : le *rappel* et la *précision*.

Le *rappel* est le rapport entre le nombre d'items pertinents sélectionnés et le nombre total d'items pertinents. Formellement :

$$R = \frac{Nb_{ps}}{Nb_p} \quad (2.7)$$

Avec :

Nb_{ps} : Nombre d'items pertinents sélectionnés

Nb_p : Nombre total des items pertinents

La *précision* est le rapport entre le nombre d'items pertinents sélectionnés et le nombre total d'items sélectionnés. Formellement :

$$P = \frac{Nb_{ps}}{Nb_s} \quad (2.8)$$

Avec :

Nb_{ps} : Nombre d'items pertinents sélectionnés

Nb_p : nombre total d'items sélectionnés

La F-mesure est un compromis entre le rappel et la précision :

$$P = \frac{2 * R * P}{R + P} \quad (2.9)$$

2.6.3 Couverture

Une autre mesure importante pour évaluer les systèmes de recommandation est la couverture des notations (Rate Coverage RC). Cette mesure fait simplement référence à la fraction de notes pour laquelle, après avoir été masquée (base de test). En d'autres termes, elle évalue si le SR est en mesure de produire une note pour l'ensemble des évaluations de test. Cette mesure est calculée par l'équation 2.10.

$$Cov = \frac{N_R}{N} \quad (2.10)$$

Avec :

N_R : est le nombre d'items pour lequel une prédiction est faite

N est le nombre d'items disponible.

2.7 Limites des SRs

Malgré leur popularité croissante et beaucoup de succès jusqu'à ce jour, les SRs rencontrent certains défis. Les systèmes de recommandation, en particulier les systèmes de recommandation basés sur le filtrage collaboratif, sont confrontés à des défis majeurs, nous citons dans ce qui suit ces défis :

- **Démarrage à froid (cold start)**

Le démarrage à froid est le problème potentiel de tout système basé sur les données, y compris les systèmes de recommandation qui tentent de créer un modèle basé sur les informations existantes. Le démarrage à froid est la situa-

tion dans laquelle l'efficacité de l'algorithme est très faible parce que le vecteur des items (ou des utilisateurs) n'a pas suffisamment d'items évalués pour trouver des vecteurs similaires. Dans l'approche basée sur le contenu, les systèmes doivent être capables de faire correspondre les caractéristiques des items aux caractéristiques pertinentes des profils des utilisateurs. Par conséquent, il doit construire un modèle détaillé des goûts et des préférences des utilisateurs. Par conséquent, sans avoir un modèle suffisamment détaillé des goûts et des préférences des utilisateurs, le système ne parviendrait pas à le faire correspondre aux items appropriés et, par conséquent, à faire une recommandation aux utilisateurs.

Dans l'approche de filtrage collaboratif, le système de recommandation identifie les utilisateurs qui partagent des préférences similaires avec l'utilisateur actif et recommande des items que les utilisateurs partageant les mêmes idées ont favorisés (et que l'utilisateur actif n'a pas encore vu). En raison du problème de démarrage à froid, cette approche ne prend pas en compte les items non évalués auparavant. Le problème de démarrage à froid peut être atténué en appliquant des approches hybrides telles qu'une combinaison d'approches de filtrage basées sur le contenu et collaboratives.

- **Rareté ou dite parcimonie (Sparsity)**

Le cœur de nombreux systèmes de recommandation est de trouver des utilisateurs similaires ou des items similaires. Bien qu'il existe de nombreux algorithmes qui peuvent résoudre ce problème, presque tous échouent lorsque la taille des vecteurs augmente et dépasse certains seuils. Lorsque le nombre d'utilisateurs ou d'items augmente, la matrice de notation devient extrêmement sparse. le tableau 2.1 montre le pourcentage de rareté de quelques dataset connus [Ayub et al., 2019].

Dans ces situations, trouver des utilisateurs similaires devient extrêmement difficile et la plupart des algorithmes existants ne parviennent pas à trouver des utilisateurs ou des items similaires. Une technique courante pour gérer ce problème consiste à utiliser des méthodes de factorisation pour réduire la taille de

Tableau 2.1: *Pourcentage de rareté dans quelques datasets*

DataSet	Pourcentage
Epinions	99.95 %
FilmTrust	98.86 %
CiaoDVD	99.90 %
MovieLens-1M	95.80 %

la matrice de notation et créer une matrice avec moins de caractéristiques plus pertinentes et indépendantes. Cependant, la gestion de matrices de notation extrêmement éparses reste un défi ouvert pour les systèmes de recommandation.

- **l'évolutivité des données**

Le nombre des utilisateurs et des items dans les systèmes de recommandation peuvent être assez importants (souvent des millions d'utilisateurs et des milliers d'items). Cela risque de ralentir la procédure de recommandation de manière significative depuis les algorithmes basés sur la mémoire pour sélectionner les plus proches voisins, ce qui exigera trop de calculs.

- **Les attaques**

Les attaques sont conçues pour amener le système de recommandation à agir de la manière souhaitée par l'attaquant. Il peut soit recommander certains items souhaités, soit empêcher de recommander d'autres items. Une attaque contre un système de recommandation consiste en un ensemble de profils d'attaque, et chacun contient des données d'évaluation biaisées associées à une identité d'utilisateur et à un élément cible.

2.8 Conclusion

Nous avons présenté dans ce chapitre, le domaine des systèmes de recommandation, les principales notions liées, ses approches et ainsi les mesures d'évaluation utilisées pour valider la performance des prédictions et des recommandations générées. Ces systèmes sont conçus pour aider les utilisateurs à trouver des ressources qui les intéressent et qui sont adaptés à leurs préférences, parmi le nombre important des choix qui s'offrent à eux.

Nous avons suivi la classification des systèmes de recommandation en trois approches principales : les approches de recommandation basées sur le contenu et les approches de filtrage collaborative ainsi hybride.

Cependant, L'intégration d'informations contextuelles dans les systèmes de recommandation s'est avéré une approche efficace pour créer des recommandations plus précises et pertinentes. Les systèmes de recommandation sensibles au contexte (les CARS : Context Aware Recommender Systems) apportent une solution à ce problème. Nous les détaillons dans le chapitre suivant.

Les Systèmes de Recommandation sensibles au Contexte (CARS)

Sommaire

3.1	Introduction	30
3.2	Définition du contexte	30
3.3	Définition de sensibilité au contexte (context-awareness)	31
3.4	Type de contexte	32
3.4.1	Contexte de l'utilisateur	33
3.4.2	Contexte de l'item	33
3.4.3	Contexte de décision	33
3.5	Terminologies utilisées	33
3.5.1	Dimensions (ou facteurs) contextuelles	34
3.5.2	Conditions contextuelles	34
3.5.3	Situations contextuelles	34
3.6	Utilisation du contexte dans les systèmes de recommandation	34
3.7	Système de recommandation sensible au contexte	35
3.8	Formalisation du contexte	36
3.9	Les approches d'incorporation du contexte	37
3.9.1	Approche de Pré-filtrage Contextuel (contextual pre-filtering)	37
3.9.2	Approche de Post-filtrage Contextuel (contextual post-filtering)	38
3.9.3	Approche de la Modélisation Contextuelle (contextual modeling)	39
3.10	Bibliothèque CarsKit	39
3.11	Synthèse sur les CARS	40
3.12	Conclusion	42

3.1 Introduction

Les systèmes de recommandation (SR) sont un moyen efficace d'alléger la surcharge d'informations en adaptant les recommandations aux préférences personnelles des utilisateurs. Les SRs avec ces différents approches supposent que les notes enregistrées d'un ensemble d'utilisateurs peuvent aider à prédire leurs évaluations inconnues. Cette hypothèse n'est valable que dans une certaine mesure. Cependant, dans de nombreuses situations, l'évaluation exacte d'un item peut être influencée par des facteurs supplémentaires et variables, appelés ici situations contextuelles. Cependant, la plupart des SRs ne feraient pas de distinction entre ces deux expériences, fournissant ainsi de mauvaises recommandations dans certaines situations.

Les systèmes de recommandation sensibles au contexte (CARS) ont été développés pour s'adapter aux préférences des utilisateurs dans différentes situations contextuelles, telles que l'heure, le lieu et la compagnie d'autres personnes (par exemple pour regarder des films).

Dans la première partie de ce chapitre, nous commençons d'abord par donner des définitions sur la notion du contexte et sensibilité au contexte, puis décrire les différentes terminologies utilisées ainsi son utilisation dans les SRs. Nous donnons la définition des systèmes de recommandation sensibles au contexte suivi des différents approches pour son intégration dans les systèmes de recommandation. Nous terminons ce chapitre par les avantages que peut apporter la bibliothèque CARSSKit sur les systèmes de recommandations sensible aux contexte et une synthèse sur ses différentes approches.

3.2 Définition du contexte

La notion de contexte a connu différentes définitions dans la littérature, chacune la définit selon ses besoins et son point de vue. Le contexte est un concept à multiples facettes qui a été étudié dans différentes disciplines de recherche, y compris l'informatique (principalement dans l'intelligence artificielle et l'informatique ubiquitaire), les sciences cognitives, la linguistique, la philosophie, la psychologie et les sciences organisationnelles. Bazire et al. dans [Bazire and Brézillon, 2005] présentent et exa-

minent 150 définitions différentes du contexte de différents domaines, et cela est dû à la complexité et du caractère multidimensionnel de ce concept. Les définitions les plus expressives sont :

Abowd [Abowd et al., 1999] donne une définition générale du contexte et le décrit comme « *Toute information pouvant être utilisée pour caractériser la situation d'une entité* » .

Dey [Dey, 2001], dans sa définition, précise les types d'entités considérées : « *Le contexte est toute information qui peut être utilisée pour caractériser la situation d'une entité. Une entité est une personne, un lieu ou un objet qui est considéré comme pertinent pour l'interaction entre un utilisateur et une application, y compris l'utilisateur et l'application eux-mêmes* » .

Chaari [Chaari et al., 2005] considère que cette définition est incomplète puisqu'elle ne précise pas ce qui compose le contexte. Ainsi, Il a proposé une nouvelle définition : « *Le contexte est l'ensemble des paramètres externes à l'application qui peuvent influencer son comportement en définissant de nouvelles vues sur ses données et ses fonctionnalités. Ces paramètres ont un aspect dynamique qui leur permet d'évoluer au cours du temps d'exécution* » .

Syvanen [Syvanen et al., 2005] développe cette propriété de contexte en déclarant que « *Le contexte est continuellement construit par la négociation entre des partenaires communicants (y compris les humains et la technologie interactive) et l'interaction d'activités et d'artefacts* » .

Nous pouvons dire que la notion de contexte n'a pas de définition précise, elle évolue en fonction de nos besoins et usages.

3.3 Définition de sensibilité au contexte (context-awareness)

Le concept de sensibilité au contexte a d'abord été introduit dans le domaine des IHM (Interactions Homme-Machine) par Weiser [Weiser, 1999] en 1991 pour réconcilier le monde virtuel et le monde physique. Il a dit que les technologies les plus

performantes sont celles associées à la vie quotidienne.

La première définition de la sensibilité au contexte a été proposée par Schilit [Schilit et al., 1993] comme *la capacité d'un système à s'adapter au contexte d'exécution en fonction de son emplacement, de tous les personnes à proximité, équipements accessibles, etc.*

Les auteurs Hull, et al. [Hull et al., 1997] le définisse comme *la capacité des appareils à détecter, ressentir, interpréter et répondre aux aspects de l'utilisateur et de l'environnement de l'appareil.*

[Ryan et al., 1998] définissent les applications sensibles au contexte comme *des applications qui surveillent les entrées de capteurs environnementaux et permettent aux utilisateurs de sélectionner les contextes en fonction de leurs intérêts et activités.*

Brown [Brown, 1998] les définit comme *les applications qui fournissent automatiquement des informations ou proposent des actions en fonction des contextes des utilisateurs détectés par les capteurs. Les actions peuvent présenter des informations à l'utilisateur pour exécuter un programme ou configurer l'interaction graphique en fonction du contexte.*

Dey [Dey, 2001] définit la sensibilité au contexte de manière plus générale : « *Un système est sensible au contexte s'il utilise le contexte pour fournir des informations et/ou des services pertinents à l'utilisateur, où la pertinence dépend de la tâche de l'utilisateur* ».

[Byun and Cheverst, 2004] donner une définition plus généralement admise : « *Un système est contextuel s'il peut extraire, interpréter et utiliser des informations contextuelles et adapter les fonctionnalités au contexte de l'usage courant* ».

Ces définitions concernent toutes la capacité d'un appareil à adapter son comportement en fonction du contexte utilisateur. [Louhab et al., 2017].

3.4 Type de contexte

Cette section présente une brève description des différents types de contexte généralement utilisés dans les systèmes de recommandation. Comme mentionné dans [29], le contexte peut être catégorisé en contexte d'utilisateur, contexte d'élément et contexte de décision. Une brève description de ces contextes est présentée dans les sous-sections suivantes.

3.4.1 Contexte de l'utilisateur

Le contexte utilisateur représente les caractéristiques contextuelles, telles que l'âge, le sexe, les amis, etc. Le contexte d'un utilisateur u est noté U_c et représenté comme l'ensemble des paires caractéristique-valeur.

Par exemple, $U_c = \langle \text{Sexe : Homme} \rangle ; \langle \text{Age : 18} \rangle$ fournit des informations contextuelles sur l'utilisateur u dont le sexe est homme et qui a 18 ans.

3.4.2 Contexte de l'item

Le contexte de l'item représente les caractéristiques contextuelles des items pour leur caractérisation. Par exemple, le genre, le sous-genre et acteurs peuvent être considérés comme des caractéristiques contextuelles des items de film. Le contexte d'un item I est noté I_c et représenté comme l'ensemble des paires caractéristique-valeur. Par exemple, $I_c = \langle \text{Genre : Sci Fi} \rangle ; \langle \text{Réalisateur : James Cameron} \rangle$ fournit des informations contextuelles sur l'item I , qui a Sci-Fi genre et James Cameron en tant que réalisateur.

3.4.3 Contexte de décision

Le contexte de décision représente les caractéristiques contextuelles, telles que l'emplacement, le compagnon et l'heure, qui sont utilisées par les utilisateurs pour l'utilisation d'un item. En fonction des valeurs de contexte d'utilisateur et d'élément, les contextes de décision changent également. Le contexte d'une décision D est noté D_c et représenté comme l'ensemble des paires de valeurs de caractéristiques. Par exemple, $D_c = \langle \text{Emplacement : Domicile} \rangle ; \langle \text{Compagnon : Enfants} \rangle$ représente un utilisateur qui a regardé un film à la maison avec ses enfants.

3.5 Terminologies utilisées

Afin de mieux comprendre la formalisation et la nouvelle structure des données contextuelle dans un système de recommandation sensible au contexte, nous présentons les différentes terminologies utilisées dans la littérature via un extrait d'une base de données, Voir figure (3.1).

User	Item	Rating	Time	Location	Compagnion
U1	I1	5	Weekend	Home	Friend
U2	I1	5	Weekday	Cinémas	Kids
U3	I2	4	Weekend	Cinema	Family
U1	I3	2	Weekday	Home	Friend
U2	I2	3	Weekend	Home	Family
U1	I2	?	Weekday	Cinema	Friend

Figure 3.1: Matrice de notation avec données contextuelles

3.5.1 Dimensions (ou facteurs) contextuelles

Sont les variables qui décrivent une situation contextuelle. Exp : Time, Location, Companion

3.5.2 Conditions contextuelles

Sont les valeurs que peut prendre une dimension contextuelle. Exp : Weekend et Weekday sont deux conditions contextuelles de la dimension Time

3.5.3 Situations contextuelles

L'ensemble des conditions contextuelles dans lesquels un utilisateur a noté un item représente une situation contextuelle. Le système tente de prédire une note à l'utilisateur $U1$ pour l'item $I2$ dans la situations contextuelle : Weekday, Cinema, Friend

3.6 Utilisation du contexte dans les systèmes de recommandation

Les systèmes de recommandation dits traditionnels utilisent deux concepts (notions) fondamentaux importants d'un système de recommandation, à savoir l'**item** et l'**utilisateur**, pour prédire les notes. Par conséquent, nous pouvons supposer qu'un système de recommandation est une fonction qui prend des utilisateurs et des items

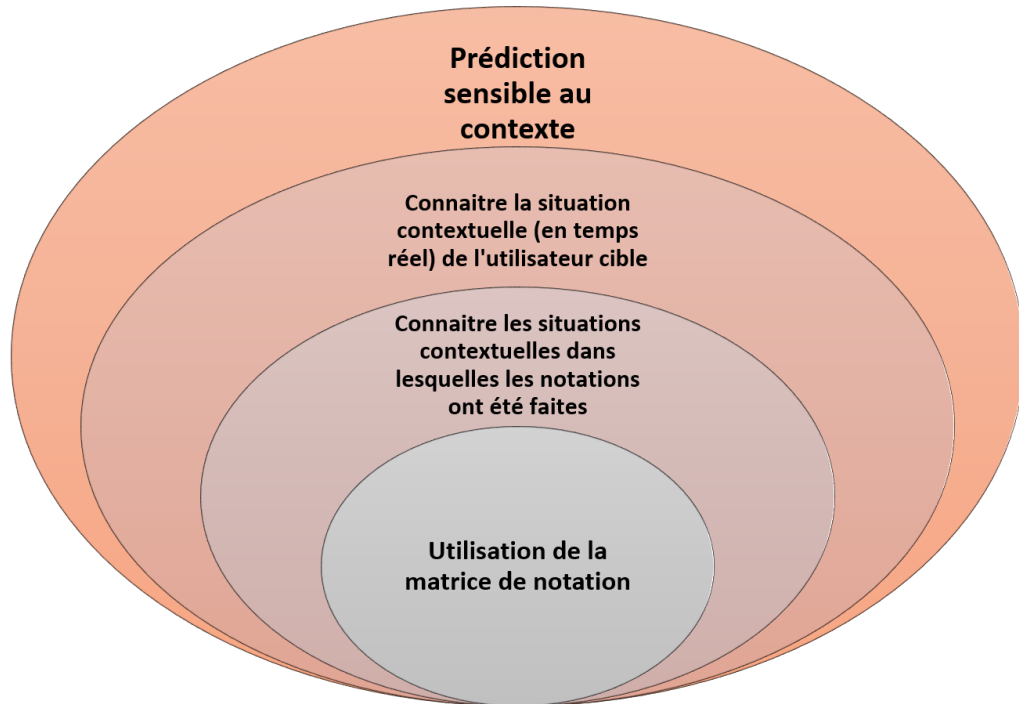


Figure 3.2: *Utilisation du contexte dans les systèmes de recommandation*

et renvoie des évaluations : $R : U \times I \rightarrow R$.

Dans cette fonction, l'entrée est en fait en 2 dimensions (2D) puisqu'elle ne prend en compte que les utilisateurs et les items. Cependant, lorsque nous ajoutons le concept de contexte dans notre système de recommandation, il devient un autre paramètre d'entrée de la fonction d'évaluation, à savoir : $R : U \times I \times C_{sit} \rightarrow R$.

Le contexte peut être considéré comme un vecteur contenant différentes informations contextuelles. Pour implémenter la méthode multidimensionnelle et tester sa performance, des données sur les items (notes) et des données contextuelles doivent être aussi collectées. Les résultats montrent empiriquement une amélioration de la prédiction des systèmes sensibles aux contexte par rapport aux systèmes qui ne les incluent pas, sachant que le calcul de prédiction prend en considération la situation contextuelle (en temps réel) de l'utilisateur cible. (voir Figure 3.2).

3.7 Système de recommandation sensible au contexte

Plus couramment nommé CARS (Context-Aware Recommender System) par la communauté [Zheng et al., 2015b]. Les systèmes de recommandation sensibles au

contexte démontrent que, selon le domaine d'application et les données disponibles, certaines informations contextuelles peuvent être utiles afin de gagner en précision dans les recommandations.

Dans CARS, les informations contextuelles sont un facteur important pour obtenir des recommandations précises. Compte tenu de l'importance de ces systèmes, plusieurs modèles de recommandation tenant compte du contexte ont été proposés.

3.8 Formalisation du contexte

Comme nous l'avons déjà mentionné, la source de données de base utilisée dans le système de recommandation est la matrice $\text{user} \times \text{item} \rightarrow \mathbb{R}$. Cette matrice contient toutes les notations, qui sont données par les utilisateurs de m pour n items. On note U l'ensemble des utilisateurs où : $U = \{u_1, u_2, \dots, u_m\}$, et par I l'ensemble des items où $I = \{i_1, i_2, \dots, i_n\}$. Si un utilisateur u évalue un article i , cela génère une note notée r_{ui} .

Les systèmes de recommandation contextuels étendent cette technologie en incorporant les informations contextuelles de l'utilisateur comme suit : $U \times I \times C_{sit} \rightarrow \mathbb{R}$, où C_{sit} représente une situation contextuelle décrivant le contexte dans lequel l'utilisateur a évalué l'item.

Formellement, le contexte est représenté comme un ensemble de dimensions contextuelles comme suit : $C = \{C_1, C_2, \dots, C_k\}$, où C_d en présente une dimension du contexte, comme le temps. Une valeur spécifique dans une **dimension contextuelle**, représente une **condition contextuelle** et est définie comme suit : $C_d = \{c_{d1}, c_{d2}, \dots, c_{dk}, \dots, c_{ds}\}$, où s est le nombre de valeurs de variante de la dimension C_d et c_{dk} est l'une des valeurs qu'une dimension peut prendre.

Par exemple, dans la dimension Temps, il existe plusieurs valeurs (telles que matin, midi, après-midi et soir) qui se présentent comme suit : $C_{time} = \{matin, midi, après-midi, soir\}$. Désormais, les données du jeu de données sont représentées dans un espace de notation multidimensionnel. Chaque note est représentée par les tuples (u, i, C_{sit}) . Par exemple : $(u_1, i_1, 5, weekend, home, friend)$ signifie u_1 notés i_1 à 5 dans la situation contextuelle : le temps est le week-end, l'emplacement est la maison et le compagnon est son ami.

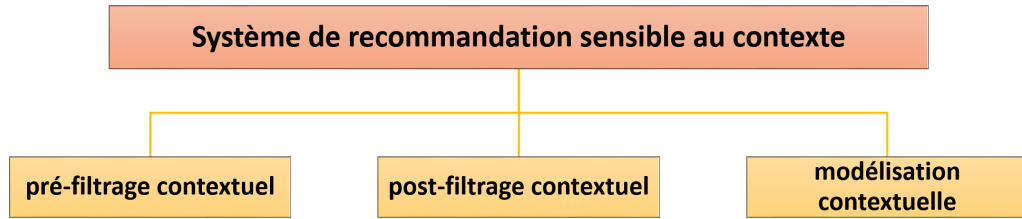


Figure 3.3: Les approches des systèmes de recommandation sensible au contexte

3.9 Les approches d’incorporation du contexte

Il est maintenant important de savoir comment et où dans le système de recommandation, les informations contextuelles vont être utilisées. Trois approches principales sont proposées pour intégrer les informations contextuelles dans la recommandation : *pré-filtrage contextuel*, *post-filtrage contextuel* et *modélisation contextuelle* présenté dans la figure 3.3 d’après : [Adomavicius et al., 2005].

La figure 3.4 illustre un aperçu général des façons dont les informations contextuelles sont incorporées dans le processus de recommandation. Le rectangle représente le processus de recommandation dans sa forme pure en séquence. Les données de notation vont dans la boîte de prédictions, qui représente le moteur qui effectue la prédiction et génère une sortie, les recommandations à la fin du processus. Le pré-filtrage contextuel filtre les données avant qu’elles n’entrent dans le moteur de prédiction. Le post-filtrage contextuel apparaît immédiatement après que le moteur de prédiction a généré sa sortie (recommandation), l’affinant pour générer une recommandation plus utile (contextuelles). La modélisation contextuelle est intégrée directement dans le moteur de prédiction. Nous les expliquerons plus en détail dans les sous sections suivantes.

3.9.1 Approche de Pré-filtrage Contextuel (contextual pre-filtering)

Le système de pré-filtrage contextuel contient un filtre de prétraitement, qui est utilisé pour contextualiser les données d’entrée des systèmes de recommandation traditionnels [Panniello et al., 2009]. Autrement dit, les informations sur le contexte

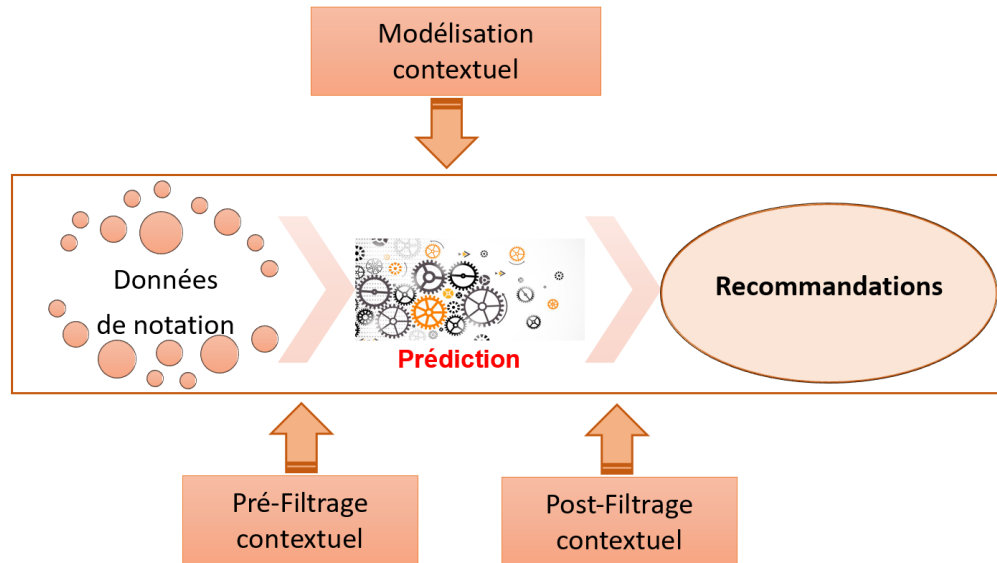


Figure 3.4: *Intégration du contexte par rapport à chaque approche*

actuel sont utilisées pour sélectionner ou construire l'ensemble de données pertinent. Ensuite, les préférences peuvent être prédites à l'aide d'un système de recommandation traditionnel sur les données sélectionnées.

De ce fait, les méthodes de recommandation traditionnelles sont utilisées pour prédire les préférences des utilisateurs en fonction de données contextualisées.

Prenons l'exemple d'un utilisateur qui souhaite trouver des hôtels dans une saison particulière ; le système de recommandation utilise uniquement les données de préférence de l'utilisateur et des autres utilisateurs pour cette saison particulière.

3.9.2 Approche de Post-filtrage Contextuel (contextual post-filtering)

Dans cette approche, les informations contextuelles ne sont prises en compte qu'à la dernière étape de la recommandation. C'est seulement à la fin que l'ensemble de recommandations résultant de ce processus est ajusté (contextualisé) pour chaque utilisateur à l'aide des informations de contexte cible. Tout comme les approches de pré-filtrage contextuel, le post-filtrage contextuel est compatible avec les algorithmes de recommandation traditionnels, avant qu'un filtre ne soit appliqué pour sélectionner les items pertinents. Les méthodes de post-filtrage peuvent fournir une réponse rapide à l'utilisateur. Les tâches de recommandation traditionnelles peuvent être effectuées

à l'avance. Lorsque des recommandations contextuelles sont nécessaires, le système peut contextualiser (filtrer ou ajuster) les résultats des recommandations.

3.9.3 Approche de la Modélisation Contextuelle (contextual modeling)

L'approche de modélisation contextuelle consiste à déployer le contexte directement dans le processus de recherche des notations inconnues. Contrairement aux méthodes de pré-filtrage et de post-filtrage, cette méthode utilise la fonction de recommandation 3D. [Liu, 2014].

3.10 Bibliothèque CarsKit

Les auteurs [Zheng et al., 2015a] mettent à la disposition des chercheurs une bibliothèque riche où les algorithmes de recommandation contextuels d'état de l'art ont été implémentés. CarsKit est open source (lien open source¹) et basé sur java. CarsKit fournit une architecture flexible permettant d'élargir facilement la portée des algorithmes de recommandations tenant compte du contexte.

Les algorithmes contextuels sont subdivisés en deux catégories dans CARSKit : *Algorithmes de transformation* et *Algorithmes d'adaptation*.

1. *Algorithmes de transformation*

Les algorithmes de transformation prétraitent les données contextuelles 3D et les convertissent en une matrice de notation bidimensionnelle contenant seulement : utilisateurs, items et notations, de sorte que tous les algorithmes de recommandation traditionnels puissent être appliqués. Ce sont les algorithmes qui se trouvent dans Baseline (Voir Figure 3.5) (utile pour cas des algorithmes de l'approche pré et post filtrage contextuelle). Parmi les algorithmes de transformation, nous trouvons les algorithmes de l'approche de découpage sensible au contexte [Zheng et al., 2014] utilisé dans notre première contribution (voir Section 6.2.1).

2. *Algorithmes d'adaptation*

Les algorithmes d'adaptation sont ceux qui incorporent directement les contextes

1. <https://github.com/irecsys/CARSKit>

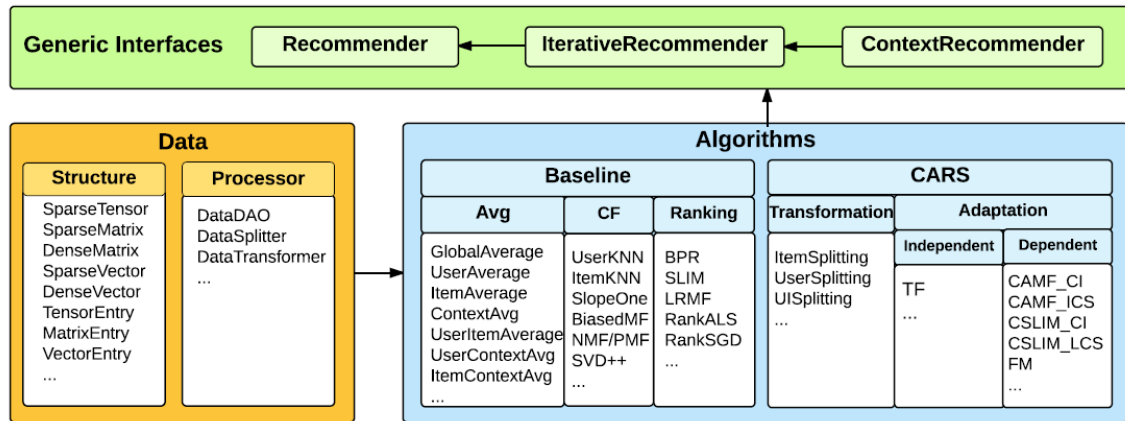


Figure 3.5: Architecture bibliothèque carskit [Zheng et al., 2015a]

dans la fonction de prédiction (cas des algorithmes de l’approche modélisation contextuelle).

Afin de pouvoir exécuter l’un des algorithmes, il faut configurer le fichier *setting.conf* (inclus dans la bibliothèque). Un guide est mis à la disposition des chercheurs [Zheng et al., 2015a] décrivant comment préparer l’ensemble de données, comment configurer les paramètres expérimentaux et comment évaluer les algorithmes, ainsi que l’interprétation des résultats. Les instructions de ce guide s’appliquent à CARSKit v0.3.5 et supérieur.

3.11 Synthèse sur les CARS

D’après les recherches, les techniques et les méthodes de CARS sont encore immatures par rapport à celles des SRs. Toutefois, les approches de recommandation traditionnelles (RS 2D) sont considérées comme des sources d’inspiration importantes pour les approches CARS, contribuant ainsi à leur développement. En conséquence, des méthodes et des techniques dans les SR 2D peuvent être empruntées et adaptées pour générer des recommandations sensibles au contexte. Aussi, bien que les approches de Pre-Filtrage et Post-filtrage de CARS puissent prendre en charge toutes les approches de recommandation 2D, ils nécessitent des étapes supplémentaires pour générer des recommandations : Pre-Filtrage doit d’abord filtrer les données non pertinentes des sources, comme étape de pré-traitement, et Post-filtrage doit filtrer les résultats de recommandation 2D non pertinents, comme étape de post-traitement. L’approche de

modélisation contextuelle nécessite des fonctions d'évaluation plus compliquées pour traiter trois dimensions ou plus de données dans le processus de génération de recommandations. Par conséquent, aucune approche de recommandation 2D ne peut être directement appliquée dans ce type d'approche, la complexité de calcul devient de plus en plus élevée.

Cependant, pour les deux autres méthodes (Pre-Filtrage et Post-filtrage), seules leurs étapes supplémentaires (pré-traitement et post-traitement) deviennent plus compliquées. Cela indique que, contrairement à Pre-Filtrage et Post-filtrage, l'approche de modélisation contextuelle est trop coûteuse lorsque le volume et la complexité des données sont importants. [Li, 2021]

Parfois, avec les méthodes de pré-filtrage contextuel, le contexte exact peut être trop étroit. Après la sélection des données dans le pré-filtrage contextuel, la rareté des données augmente. Par conséquent, le système peut ne pas avoir suffisamment de données pour faire des recommandations précises [Adomavicius and Tuzhilin, 2011].

Prenons l'exemple d'une recommandation de films : une personne veut voir un film avec famille dans un théâtre, le vendredi soir. Si le contexte exact est utilisé, il est probable que les résultats ne soient pas précis et fiables. Premièrement, peut-être pas suffisamment de données pouvant correspondre au contexte exact. Dans ce cas, il est possible aussi de n'avoir aucune recommandation pour ce contexte très spécifique. Des solutions ont été proposées pour sélectionner juste quelques dimensions [Zheng et al., 2012] ou toutes les dimensions mais pondérées [Zheng et al., 2013].

Le post-filtrage contextuel est confronté aux mêmes problèmes. En fait, les problèmes de généralisation du contexte sont partagés par toutes les méthodes de contextualisation. Un contexte précis et détaillé peut à nouveau conduire à la rareté des données ; tandis que le contexte généralisé conduit à des recommandations inexactes [Liu, 2014]. Enfin, nous constatons que les travaux de SR améliorés par l'information contextuelle ne résolvent pas les problèmes de SR, plus particulièrement démarrage à froid et rareté.

Les chercheurs ont tourné leur attention vers l'utilisation d'autres informations

supplémentaires tel que : les liens de confiance entre les utilisateurs pour remédier ces limites. Ainsi, exploiter l'apport des technologies du Web sémantique en utilisant des informations sémantiques décrivant les utilisateurs et les ressources à travers les données liées.

3.12 Conclusion

Nous avons discuté dans ce chapitre sur l'importance du contexte dans le domaine des systèmes de recommandation. Nous avons d'abord présenté dans la première partie la notion du contexte dans les systèmes de recommandation. Puis, nous avons étalé les terminologies utilisées dans le domaine des SRs. Puis nous avons détaillé les différentes approches utilisées pour intégrer l'information contextuelle. Afin d'exploiter les travaux d'état de l'art et les comparer avec les nouvelles propositions, nous avons exposé et expliqué l'utilisation et paramétrage de la bibliothèque CarKit utilisée dans la partie expérimentation de notre travail. Nous avons clôturé ce chapitre par l'ensemble des problèmes et limites rencontrés par les systèmes de recommandation sensible au contexte, d'où la nécessité d'intégrer d'autres informations.

Les systèmes de recommandation basés sur la confiance et la sémantique

Sommaire

4.1	Introduction	44
4.2	Web Social	44
4.3	Web sémantique	45
4.4	Confiance (Trust)	46
4.4.1	Web Social et confiance	46
4.4.2	Définitions	47
4.4.2.1	Confiance	47
4.4.2.2	Estimation de la confiance	47
4.4.2.3	Réseau de confiance	49
4.4.3	Propriétés de la confiance	50
4.4.4	Formalisation du réseau de confiance	52
4.5	Sémantique et données ouvertes liées	53
4.5.1	Mesures de similarité sémantique pour SR	54
4.5.1.1	Objectif des Similarités sémantiques	54
4.5.1.2	Approches de mesure de similarité sémantique	55
4.6	Systèmes de recommandation sensibles à la confiance	56
4.6.1	Architecture TARS avec confiance explicite et implicite	56
4.6.1.1	Architecture avec Confiance explicite	57
4.6.1.2	Architecture avec Confiance implicite	58
4.6.2	Métriques de confiance dans TARS	58
4.7	Systèmes de recommandation basé sur la sémantique	59
4.8	Conclusion	60

4.1 Introduction

Les Systèmes de recommandations sont des techniques bien établies pour gérer avec succès la surcharge d'informations due à la croissance des données sur le Web. Les CARS se sont avérées fiables pour fournir des prédictions plus pertinentes et plus précises en incorporant les situations contextuelles de l'utilisateur. Bien que le filtrage collaboratif (FC) soit la technique largement utilisée et la plus efficace pour les CARS, mais il souffre toujours des issues de SR traditionnel .

Plusieurs propositions restent ouvertes pour d'autres améliorations afin de diminuer ces limites. Dans ce chapitre, Nous tentons d'introduire deux types de travaux dans les CARS pour résoudre ces problèmes tout en maintenant la qualité des recommandations ; Principalement les SR basé confiance (TARS) puis ceux basés sur la sémantique (SARS) et leurs impacts sur les CARS.

D'un côté, l'émergence des réseaux sociaux et leur croissance drastique suggère que la quantité d'informations qu'ils contiennent pourraient être utiles dans de nombreuses applications, y compris les systèmes de recommandation. Sachant qu'en apprenant des nouveaux types de données extraites des réseaux sociaux, nous pouvons aider un système de recommandation à mieux trouver des utilisateurs similaires.

D'un autre côté, l'utilisation des informations sémantiques décrivant les utilisateurs et les items améliorent la précision et la qualité de recommandation. Sachant que l'exploitation des technologies du web sémantique permet d'améliorer la qualité de la recommandation.

Dans ce chapitre, nous allons présenter les concepts de base de la confiance et sémantique , puis nous définissons les deux types de système de recommandation : CARS et SARS pour décrire comment incorporer ces informations dans les SRs.

4.2 Web Social

Le Web n'a cessé d'évoluer depuis sa création, d'abord le Web 1.0 était une grande bibliothèque avec un grand nombre de documents où les utilisateurs pouvaient obtenir

de l'information. Cette information était à sens unique, jusqu'à la venue du Web 2.0. Ce dernier a été introduit par en 2004 qui a complètement changé la vision du Web, en plaçant l'internaute au coeur de l'information, ainsi son rôle a évolué d'un simple récepteur passif d'information à un contributeur actif. En d'autres termes, le Web 2.0 peut être décrit comme l'ensemble des technologies et applications Web favorisant les échanges et rendant le Web un véritable espace de communication. C'est ce qui fait que le Web 2.0 est aussi synonyme du Web social, Web collaboratif, ou Web communautaire etc.

D'un côté, le Web 2.0 a pu maximiser le potentiel de création et de partage et d'interactions des internautes avec les contenus du Web , et d'un autre côté, il a pu mettre en avant les échanges sociaux entre les utilisateurs. Ces révolutions techniques et technologiques du Web 2.0 ont donné naissance à de nouveaux outils de communication dont les plus répandus sont les réseaux sociaux.

4.3 Web sémantique

Le Web sémantique est une nouvelle façon d'utiliser Internet pour lier un ensemble de documents. Il s'agit d'arriver à un Web intelligent, où les informations ne seraient pas juste stockées mais comprises par les machines afin d'apporter des réponses pertinentes à l'utilisateur ; également connu sous le nom de Web 3.0 (voir Figure 4.1). Il est introduit par Tim Berners-Lee dans les années quatre-vingt-dix.

[Berners-Lee et al., 2001] synthétise le Web sémantique comme étant un Web où des ordinateurs utilisant des agents intelligents qui analysent toutes les données sur le Web (contenu, liens, transactions), qu'elles soient en langage naturel ou non. En d'autres termes « on passe d'un Web 2.0 qui exploite l'intelligence collective des hommes à un Web qui exploite l'intelligence collective des capteurs et des données ».

Le Web sémantique entend alors remplacer le "Web des documents" par le "Web des données" d'où l'appellation Web de données qui veut également désigner le Web sémantique. Plus concrètement l'infrastructure du Web sémantique identifie et transforme les ressources de manière vigoureuse tout en renforçant l'aspect d'ouverture de données. Elle doit également assurer l'interopérabilité et faciliter la mise en oeuvre des calculs et des raisonnements complexes tout en maximisant leur validité.

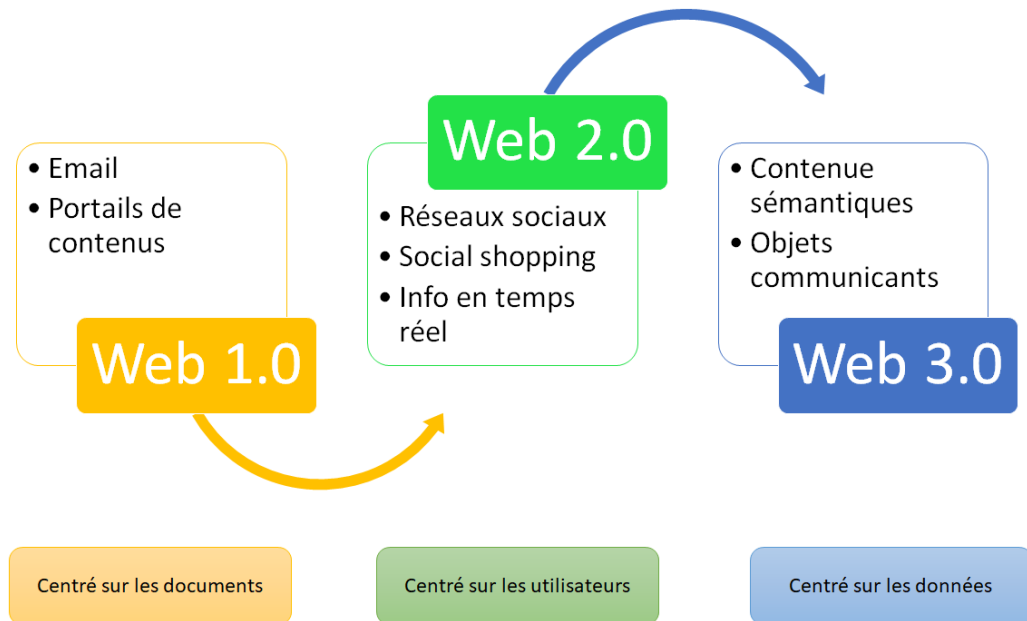


Figure 4.1: *Évolution du Web*

4.4 Confiance (Trust)

4.4.1 Web Social et confiance

Le développement du Web 2.0 a permis aux utilisateurs de partager des informations en ligne, par exemple avec les réseaux sociaux, l'affichage d'informations personnelles est devenue une pratique courante; les questions de confidentialité et de protection des données personnelles divisent les membres des réseaux sociaux et autres internautes. Un réseau social pair à pair considère que chaque membre du réseau social est associé à un pair et qu'il est lié aux autres pairs par des liens d'amitié et/ou de confiance définis manuellement.

La confiance joue un rôle clé dans le processus de prise de décision d'une personne. Dans le monde réel, les gens sont prêts à recevoir uniquement les conseils et les recommandations de leurs amis dignes de confiance .

L'incorporation d'informations de confiance dans les SRs, aboutit à une nouvelle classe de systèmes de recommandation appelés systèmes de recommandation sensibles à la confiance (TARS).

Cette thèse examine l'apport de la modélisation et de l'exploitation de la confiance aux SRs et plus particulièrement aux CARS. Sachant que les méthodes de recommandation basées sur la confiance tentent d'exploiter les informations sur les relations

sociales des réseaux de confiance des utilisateurs pour améliorer les performances des SR et plus particulièrement Filtrage collaborative, où les voisins de confiance peuvent être utilisés pour prédire les nouveaux items non notés pour les recommander aux utilisateurs.

4.4.2 Définitions

4.4.2.1 Confiance

Dans la littérature, il existe des dizaines de définitions, dans différents contextes et situations, et des variations considérables dans la signification de la confiance, comprenant la sociologie, psychologie, sciences économiques et informatique. Chacune de ces disciplines a définie et considéré la confiance de différentes perspectives.

Nous retenons dans notre travail, les définitions suivantes. La première bien expressive mais générique [Massa, 2007] : *« l'opinion explicite exprimée par un utilisateur sur un autre utilisateur concernant la qualité perçue d'une certaine caractéristique de cet utilisateur »*. Aussi, nous nous basons sur la définition citée par [Haydar, 2014], qui permet de prendre en compte les perspectives des SRs : *« La confiance est la croyance d'un individu nommé confiant, qu'un autre nommé crédible (ou digne de confiance), a la compétence et la volonté de coopérer, pour accomplir une tâche en faveur du confiant. »*

La confiance est largement acceptée comme un composant important dans les relations sociales. En général, la confiance est la mesure qu'une entité se comportera comme prévue, en dépit du manque de capacité de surveiller ou commander l'environnement dans lequel elle fonctionne.

4.4.2.2 Estimation de la confiance

L'estimation de la confiance envers les utilisateurs est faite selon deux facteurs : Appréciation des intentions à l'égard du confiant, ainsi l'appréciation de la compétence des autres utilisateurs à propos de la tâche à effectuer.

Selon [Marsh and Briggs, 2009], l'estimation de la confiance par le confiant est faite selon le classement dans la barre de confiance. Cette dernière est subdivisée en trois zones. La figure montre la barre de confiance : la confiance varie entre deux valeurs -1 (défiance complète) et +1 (confiance complète). La valeur 0 représente la

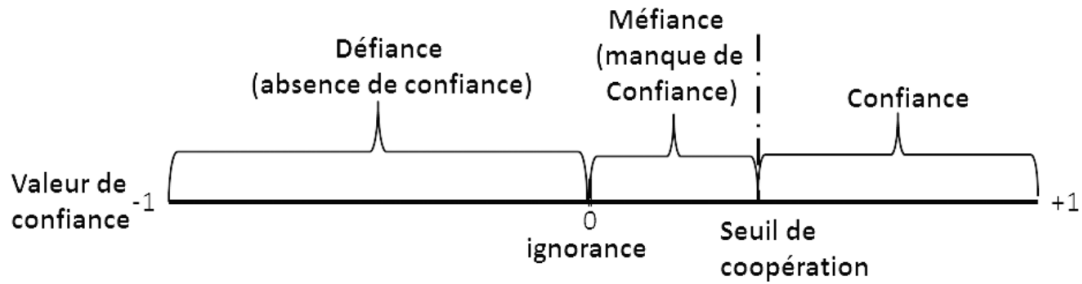


Figure 4.2: Zones de confiance [Haydar, 2014]

neutralité. Le seuil de coopération est la valeur minimale de confiance pour laquelle le confiant accepte de coopérer avec un autre.

On peut identifier trois zones :

1. **La défiance** : la valeur est négative parce que le confiant pense que l'utilisateur a des intentions négatives.
2. **La méfiance** : l'estimation du confiant est positive, les intentions l'utilisateur ne présentent pas de danger pour lui, mais la valeur reste en-dessous du seuil de coopération. Cela parce que le confiant juge l'utilisateur est incapable d'exécuter la tâche, même si ses intentions sont positives. Par conséquent, le confiant ne va pas coopérer avec lui tant qu'il émet des doutes sur sa compétence ou ses avis.
3. **La confiance** : la valeur ici est au-dessus du seuil de coopération, alors le confiant prendra le risque de coopérer avec cet utilisateur.

La valeur de confiance (représenté par : t_u, v) peut être soit des nombres binaires, soit des nombres réels (c'est-à-dire dans la plage de $[0, 1]$). La confiance binaire est le moyen le plus simple d'exprimer la confiance. Soit deux utilisateurs se font confiance ou non. Une méthode plus compliquée est le modèle de confiance continue, qui attribue des valeurs réelles aux relations de confiance. De nombreuses applications, telles qu'Amazon¹ et eBay², utilisent des valeurs de confiance binaires

Pas mal de travaux qui intègrent à la fois la confiance et la méfiance pour améliorer la qualité des systèmes de recommandation sensible à la confiance ([Ma et al.,

1. www.amazon.com

2. www.ebay.com

2009], [Lee and Ma, 2016], [Richa and Bedi, 2021]) . Les informations de méfiance prennent en compte les désaccords entre les utilisateurs. Alors que les informations de confiance montrent à quel point un utilisateur aime les recommandations faites par un autre utilisateur. Les informations de méfiance montrent le contraire.

En tenant compte des informations de méfiance, les systèmes de recommandation peuvent éviter d’obtenir des recommandations d’utilisateurs dont l’utilisateur actuel se méfie. Les informations sur la méfiance aident également les SRs à découvrir des items que l’utilisateur n’aime pas. Cela vient de l’idée que si un utilisateur ne fait pas confiance à un autre utilisateur, il n’aime pas les items recommandés par l’utilisateur. Ainsi, le système de recommandation évitera également de les recommander [Abbasi et al., 2014].

4.4.2.3 Réseau de confiance

L’émergence des réseaux sociaux et leurs croissance apportent d’énormes informations. Ces dernières pourraient être utiles dans de nombreuses applications, y compris les systèmes de recommandation.

L’utilisation des réseaux de confiance a le même principe que le filtrage collaboratif mais qui pousse plus loin la similarité entre les utilisateurs. Les recherches sur les réseaux de confiance sont motivées par l’hypothèse qu’il existe d’autres facteurs, que les mesures de similarité traditionnelles utilisées dans le filtrage collaboratif, qui peuvent influencer un utilisateur à acheter un produit par exemple.

L’intérêt principal des réseaux de confiance est la **relation de confiance** qui relie les utilisateurs entre eux. Contrairement au filtrage collaboratif où la similarité se fait seulement par rapport aux préférences des utilisateurs.

Le principal avantage d’ajouter un réseau de confiance à un système utilisant le filtrage collaboratif est que celui-ci permet d’assouplir le problème du démarrage à froid et de rareté des données [Massa and Avesani, 2004], [Guo et al., 2012], [Gupta and Nagpal, 2015]. En d’autres termes, avec le réseau de confiance, même si l’utilisateur est nouveau dans le système, il pourra quand même recevoir des recommandations s’il est relié à d’autres individus. Évidemment, cela nécessite que l’individu connaisse d’autres utilisateurs qui font partie de ce système.

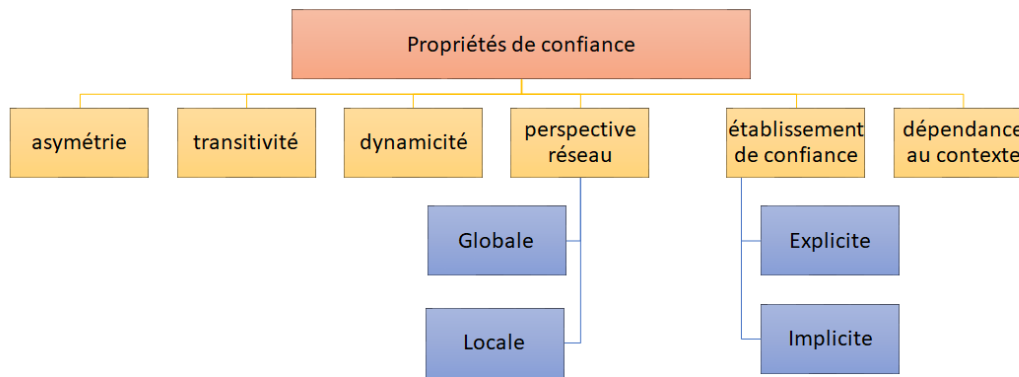


Figure 4.3: *Propriété transitivité de la confiance*

[Lee and Brusilovsky, 2009] dans leurs article baptisé : "Does Trust Influence Information Similarity ?" montre que les utilisateurs connectés par un réseau de confiance présentent une similitude significativement plus élevée sur les items que les utilisateurs non connectés. Cette similitude est la plus élevée pour les utilisateurs directement connectés et diminue avec l'augmentation de la distance entre les utilisateurs.

4.4.3 Propriétés de la confiance

Plusieurs propriétés de confiance ont été définies, notamment : Asymétrie, Transitivité, Dynamicité, Propagation, perspective réseau, établissement de confiance et dépendance au contexte [Gupta and Nagpal, 2015] (Voir Figure 4.3).

1. Asymétrie (asymmetry)

La confiance est une relation asymétrique et personnelle entre les utilisateurs ; par conséquent, il crée une relation dirigée dans les réseaux sociaux. En d'autres termes, si $t_{u,v}$ représente la valeur de la confiance que u a en v , elle pourrait ne pas être égale à la valeur de la confiance que v a en u . Par conséquent, la confiance est dirigée et asymétrique.

$$t_{u,v} \neq t_{v,u} \quad (4.1)$$

2. Transitivité(transitivity)

La transitivité est une propriété clé de la confiance. Sur la base de l'effet de

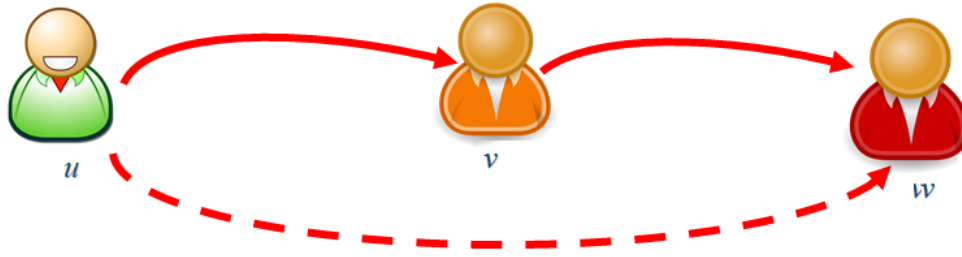


Figure 4.4: *Propriété transitivité de la confiance*

transitivité, si u fait confiance à v et v fait confiance à w , on peut en déduire que u pourrait également faire confiance à w dans une certaine mesure,

$$t_{u,v} \wedge t_{v,w} \implies t_{u,w} \quad (4.2)$$

ça permet à la confiance de se propager le long des chemins pour atteindre d'autres utilisateurs (Voir Figure 4.4).

3. **Dynamicité (dynamicity)**

La confiance est une quantité dynamique. Il change continuellement avec le temps. Les faits positifs augmentent la confiance tandis que les faits négatifs diminuent la confiance.

4. **Propagation(propagation)**

La propagation est une propriété de confiance que les SR profite dans le processus de prédiction du score de confiance via des chemins de confiance connus. Les relations de confiance directes dans le réseau de confiance d'un utilisateur créent un chemin par lequel de nouvelles connexions indirectes peuvent être établies avec d'autres utilisateurs. Dans le passé, les chercheurs ont incorporé diverses méthodologies pour la propagation de la confiance.

5. **Perspective réseau (network perspective)**

Indique Confiance globale (Global Trust) et confiance locale (Local Trust)

- confiance locale : se base sur la mesure d'un utilisateur à un autre utilisateur.

En d'autres termes, il s'agit du degré d'une relation de confiance entre deux utilisateurs, Il est local dans le sens où il est calculé entre chaque paire d'utilisateurs, plutôt que basé sur tous les utilisateurs du réseau.

- La confiance globale, quant à elle, est le consensus de l'ensemble de la com-

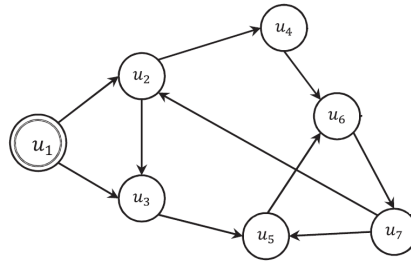


Figure 4.5: Exemple représentation d'un réseau de confiance

munauté sur la fiabilité d'un utilisateur. La confiance globale est la réputation qu'un utilisateur a dans le réseau. Dans la littérature sur les systèmes de recommandation sensibles à la confiance, la métrique de confiance locale est généralement préférée bien qu'il existe des systèmes qui adoptent à la fois la confiance locale et globale tels : [Tang et al., 2013], [Sejwal and Abulaish, 2019].

6. **Établissement de confiance(trust establishment)**

Cette propriété indique la confiance explicite ou implicite : l'établissement de la confiance peut être basé sur des réseaux de confiance explicites ou implicites. Les réseaux explicites sont construits avec des déclarations de confiance explicites, qui sont directement fournies par un utilisateur pour un autre utilisateur. Alors que les déclarations de confiance implicites sont déduites du comportement de l'utilisateur. Les relations de confiance implicites sont calculées généralement via la similarité des utilisateurs et d'autres mesures de confiance.

7. **Dépendance au contexte(context dependency)**

La confiance dépend du contexte. Cela signifie que faire confiance à quelqu'un sur un sujet ne garantit pas également de lui faire confiance sur d'autres sujets. Par exemple, une personne digne de confiance en technologie peut ne pas être digne de confiance en astronomie.

Ces propriétés sont extraites de la définition de la confiance et fournissent la base pour la création d'algorithmes qui utilisent les informations de confiance.

4.4.4 Formalisation du réseau de confiance

Le réseau de confiance est un graphe pondéré et orienté dans lequel les nœuds sont des utilisateurs et les arcs sont la confiance entre les utilisateurs.

	u_1	u_2	u_3	u_4	u_5	u_6	u_7
u_1	1	1	1	-	-	-	-
u_2	-	1	1	1	-	-	-
u_3	-	-	1	-	1	-	-
u_4	-	-	-	1	-	1	-
u_5	-	-	-	-	1	1	-
u_6	-	-	-	-	-	1	1
u_7	-	1	-	-	1	-	1

Figure 4.6: Exemple représentation de Matrice des données de confiance

Les arcs expriment le degré de confiance entre les utilisateurs aux deux extrémités de chaque bord.

L'utilisateur actif, u peut identifier un ensemble d'utilisateurs de confiance, désigné par T_u . Si l'utilisateur u fait confiance à l'utilisateur v , la valeur de la confiance est exprimée en $t_{u,v}$. C'est un nombre réel dans l'intervalle $[0, 1]$.

La matrice de confiance $T : U \times U \rightarrow [0, 1]$ (Voir Figure 4.6) peut être utilisée pour décrire la structure du réseau de confiance. T peut ne pas être symétrique car u fait confiance à v ne signifie pas toujours que v fait confiance à u de la même manière. La figure 4.5 représente un réseau de confiance d'un utilisateur u_1 utilisant des valeurs de confiance binaire.

4.5 Sémantique et données ouvertes liées

A partir de 2006, Tim Berners-Lee rappelle que le Web sémantique n'a pas vocation uniquement à poser des données dans le Web, mais à relier les données directement entre elles pour qu'une machine ou un humain puisse explorer le Web de données (également appelées données liées). Le Web de données acquiert immédiatement une base de travail solide mais aussi une bonne visibilité auprès des spécialistes du Web et de son évolution.

En tirant parti des normes et technologies du Web sémantique, une quantité croissante de contenu sémantique a été publiée sur le Web sous forme de données ouvertes liées (Linked Open Data (LOD)) [Meymandpour and Davis, 2016].

Les données liées (linked data) désignent des données suivant un paradigme fondé sur quatre règles simples : les URIs pour identifier les entités, les URLs permettant le référencement des entités, fournissant des informations utiles à ces URI fondées sur des formats standard et la connexion et l'interconnexion à d'autres entités afin de permettre une exploration plus approfondie

Elles permettent de publier des données structurées de manière standard afin que des ensembles de données de divers domaines puissent être liés entre eux.

LOD est un récent effort communautaire qui donne accès à une quantité importante et croissante de données structurées diverses en utilisant les normes ouvertes du Web sémantique et via des licences libérales. Le contenu sémantique de LOD en conjonction avec les mécanismes avancés de recherche et d'interrogation fournis par SPARQL a ouvert des opportunités, non seulement pour améliorer les applications existantes, mais aussi pour développer de nouvelles applications sémantiques intelligentes innovantes. l'idée du Linked Data connaît sa plus importante réalisation dès février 2007 avec la création de Dbpedia [Bizer et al., 2009].

DBpedia (dans le cadre de LOD) est une base de connaissances accessible au public extraite de Wikipédia sous la forme d'une ontologie de concepts et de relations, rendant cette vaste quantité d'informations accessible par programmation sur le Web. Ces connaissances librement disponibles ont été utilisées dans plusieurs travaux pour améliorer la qualité des systèmes de recommandation [Ostuni et al., 2013] , [Peska and Vojtas, 2013] , [Maleček et al., 2019].

4.5.1 Mesures de similarité sémantique pour SR

4.5.1.1 Objectif des Similarités sémantiques

L'objectif des mesures de similarité sémantique est d'évaluer la proximité sémantique entre les concepts. Le calcul de similarité entre deux concepts permet de déterminer s'ils sont similaires c'est-à-dire s'ils atteignent un certain niveau de ressemblance ou dissimilaire qui peuvent être également liés sémantiquement par des relations lexicales : antonymie, spécialisation, etc.

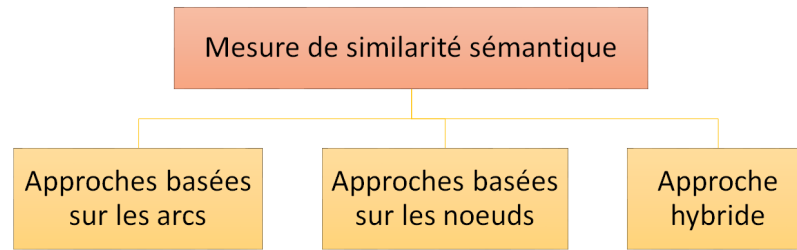


Figure 4.7: *Approche de mesure de similarité sémantique*

4.5.1.2 Approches de mesure de similarité sémantique

Nous distinguons trois grandes catégories de mesures de similarité sémantique.

- Approches basées sur les arcs

La mesure de similarité la plus intuitive des objets dans une ontologie est leurs distances. Cette similarité est évaluée par la distance qui sépare les objets dans l'ontologie. Ces mesures se servent de la structure hiérarchique de l'ontologie pour déterminer la similarité sémantique entre les concepts. Le calcul des distances dans l'ontologie est basé sur un graphe de spécialisation des objets. Dans chaque graphe, la distance de l'ontologie doit être caractérisée par le plus court chemin qui fait intervenir un ancêtre commun ou le plus petit généralisant, connectant potentiellement deux objets à travers des descendants communs. Parmi les travaux classifiés sous cette catégorie on peut citer : Mesure de Wu and Palmer [Wu and Palmer, 1994], Mesure de Rada [Rada et al., 1989]

- Approches basées sur les noeuds

Ces approches adoptent une nouvelle mesure en termes de la mesure entropique (Contenu Informationnel) de la théorie de l'information. La probabilité P pour l'identification de l'utilisation d'une classe ou de ses descendants dans un corpus désigne l'information de la classe. En associant des probabilités aux concepts d'une taxonomie, il est possible d'éviter le manque de fiabilité des distances des arcs. Cette caractéristique quantitative de l'information fournit une nouvelle façon de mesurer la similarité sémantique. Plus l'information est partagée par deux concepts, plus ils sont similaires. Parmi les travaux, recensés dans la littérature, sous cette catégorie on peut citer : Resnik [Resnik, 1995],

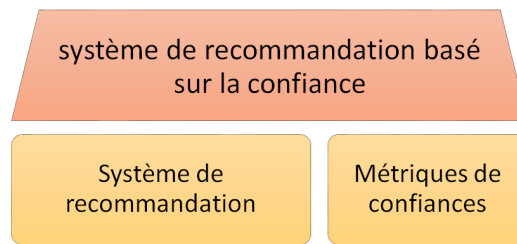


Figure 4.8: *Piliers du Système de recommandation basé sur la confiance*

Mesure de Lin [Lin et al., 1998].

- Approches hybrides

Ces approches sont fondées sur un modèle qui combine entre les approches basées sur les arcs (Distances) en plus du contenu informationnel qui est considéré comme facteur de décision. Parmi ces approches on cite : Mesure de Jiang et Conrath [Jiang and Conrath, 1997], Mesure de Leacock et Chodorow [Leacock and Chodorow, 1998]

4.6 Systèmes de recommandation sensibles à la confiance

Baptisé aussi Systèmes de recommandation basé sur la confiance ; connu en anglais sous : Trust-Aware Recommender System (TARS). TARS sont des systèmes qui intègre les informations de confiance dans les systèmes de recommandation. Plus précisément, les systèmes de recommandation et les métriques de confiance sont les deux piliers du système de recommandation basé sur la confiance (Voir Figure 4.8).

Les métriques de confiance sont les méthodes de mesure et de calcul de la valeur de la confiance entre les utilisateurs du réseau.

4.6.1 Architecture TARS avec confiance explicite et implicite

Les relations de confiance peuvent être envisagées sous différents angles. La nature de l'information de confiance (explicite ou implicite) influe sur l'architecture d'un système TARS.

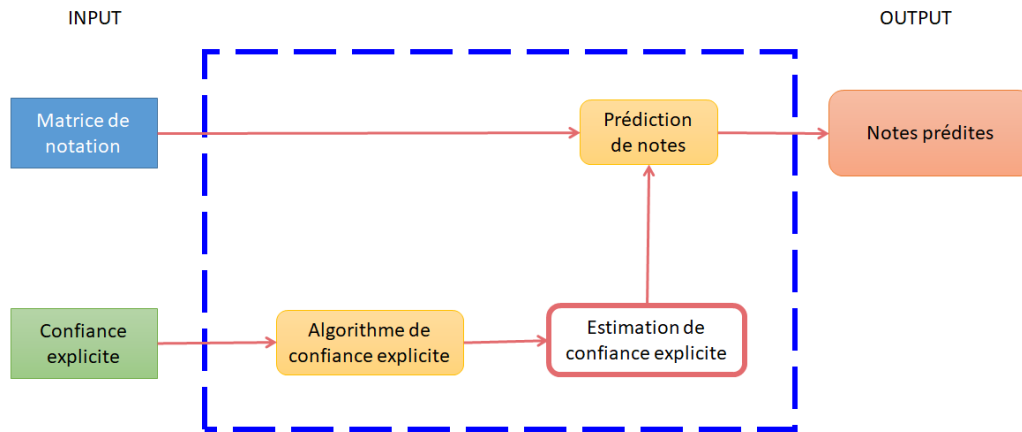


Figure 4.9: Architecture TARS basée sur la confiance explicite

4.6.1.1 Architecture avec Confiance explicite

Comme déjà mentionné dans la partie "propriétés des TARS", on parle de confiance explicite quand les déclarations de confiance sont spécifiées par les utilisateurs eux-mêmes. Elles sont utilisées comme entrées de TARS avec les notes sur les items pour prédire les notes manquantes.

Bien que les TARS basés sur la confiance explicite a été vérifié pour avoir une précision de prédiction, ils ont quelques limites à savoir : il est parfois long ou coûteux d'obtenir la confiance explicite. En effet, la confiance explicite nécessite des efforts supplémentaires de la part des utilisateurs [Massa and Avesani, 2009] : les utilisateurs doivent indiquer spécifiquement leurs opinions personnelles sur les utilisateurs dignes de confiance.

La figure 4.10 montre l'architecture du TARS se basant sur la confiance explicite [Gupta and Nagpal, 2015] .

Par exemple, les utilisateurs du site Web opinion³ peuvent ajouter d'autres utilisateurs à leur «liste de confiance» et former un «Web de confiance» ; de même Ciao.co.uk⁴ adopte un concept similaire appelé «Cercle de confiance» dans lequel les utilisateurs sont considérés comme dignes de confiance. Ce dernier sera utilisé dans la partie expérimentation de notre travail.

De nombreux systèmes de recommandation basés sur la confiance explicites ont été proposés dans la littérature, dont nous citons : [Guo et al., 2015], [Azadjalal et al.,

3. <https://www.epinions.com>

4. <https://www.ciao.com/>

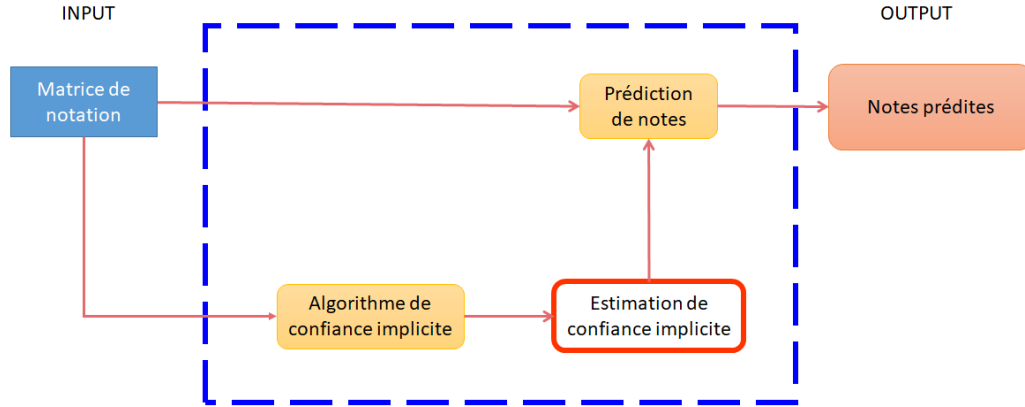


Figure 4.10: Architecture TARS basée sur la confiance implicite

2017], [Yadav et al., 2018] et [Sejwal and Abulaish, 2019]

4.6.1.2 Architecture avec Confiance implicite

Les systèmes TARS basés sur la confiance implicite font référence aux informations de confiance implicitement déduites du comportement de l'utilisateur dans le système, dû au limites des TARS utilisant la confiance explicite qui nécessitent des efforts supplémentaires de l'utilisateur pour étiqueter les déclarations de confiance. Des travaux existants utilisent le réseaux de confiance implicite au lieu d'utiliser la confiance explicite [Yadav et al., 2018], aussi [Yuan et al., 2010] utilise les informations de similarité utilisateur facilement disponibles pour générer les confiances implicites pour TARS.

4.6.2 Métriques de confiance dans TARS

Les techniques ou métriques de confiance sont très similaires à celles du FC traditionnel. Si la prédiction dans chaque recommandation basé FC traditionnel est basée sur la similarité de l'utilisateur actif. La recommandation dans les systèmes TARS [Massa and Avesani, 2007a] incluait les voisins de confiances de l'utilisateur actif, qui peut être définie comme dans l'équation suivante (4.3).

$$p_{u,i} = \bar{r}_u + \frac{\sum_{v \in TN_u} t_{u,v} (r_{v,i} - \bar{r}_v)}{\sum_{v \in TN_u} t_{u,v}} \quad (4.3)$$

Avec :

$t_{u,v}$: valeur de confiance entre l'utilisateur u and v

TN_u : ensemble de voisins de confiance de l'utilisateur u ayant noté l'item i ,

$r_{v,i}$: note attribuée par l'utilisateur v à l'item i

Nous pouvons classer ces métriques en deux catégories, celles basées sur la confiance explicites et celles basées sur la confiance implicite [Yadav et al., 2018].

Massa et Avesani [Massa and Avesani, 2007a] propose une métrique. Cette dernière part du principe que la fiabilité de la confiance propagée diminue avec chaque nouvelle étape de propagation de la confiance. Par conséquent, pour garantir que la confiance diminue au cours de la propagation, ils utilisent la métrique de confiance MoleTrust (utilisée dans nos contributions).

$$t_{u,v} = \left(\frac{d_{max} - d_{u,v} + 1}{d_{max}} \right) \quad (4.4)$$

Avec :

$d_{u,v}$: la plus courte distance de propagation de la confiance entre utilisateur u et le confiant v

d_{max} : la distance de propagation maximale (maximum allowable propagation distance (MAPD)), qui représente toute valeur entière positive.

La distance de propagation de confiance fait référence au nombre de sauts dans le chemin de propagation de confiance le plus court entre le confiant et le crédible.

Des détails sur les métriques existantes sont citées dans : [Haydar, 2014], [Yadav et al., 2018].

4.7 Systèmes de recommandation basé sur la sémantique

Ces dernières années, des techniques de raisonnement empruntées du Web sémantique ont été adoptées dans les systèmes de recommandation pour atténuer avec succès le problème de rareté et d'autres faiblesses récurrentes auxquelles ils sont confrontés depuis leur création. Nous trouvons dans la littérature ce nouveau type de système, connu en anglais sous : Semantic-Aware recommender system (SARS). Nous pou-

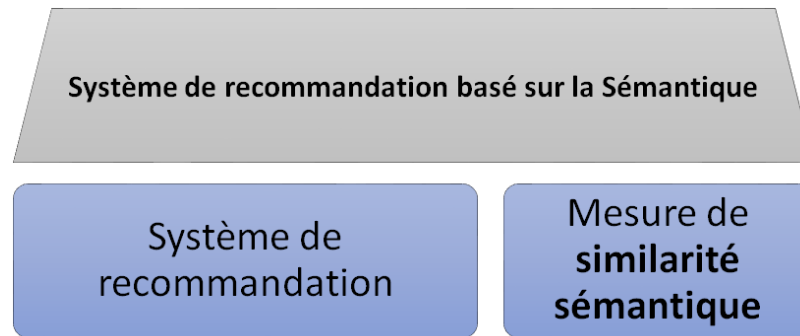


Figure 4.11: *Les piliers des Systèmes de recommandation basé sur la sémantique*

vons dire que les deux piliers de ce système sont les systèmes de recommandation et l'ensemble des mesures de similarité sémantique (Voir Figure) 4.11.

De ce fait, l'utilisation de taxonomies et d'ontologies de produits permet au système de raisonner sur la sémantique des items et de découvrir les associations sémantiques cachées entre eux. autrement dit, les items et/ou utilisateurs d'un système de recommandation peuvent être représentés par des instances de concepts définis dans l'ontologie du domaine ciblé par la recommandation. La mesure de similarité entre ces ressources au travers de leur caractérisation sémantique pourra être adoptée.

Aussi, comme les LOD pour Linked Open Data, représentent un gigantesque graphe de données et de connaissances, présent sur Internet, structuré par de multiples connexions entre ces ressources, des travaux proposent de tirer parti des technologies liées à ce paradigme pour enrichir l'approche de filtrage collaboratif lors des démarrages à froid.

4.8 Conclusion

Les SRs sont des systèmes intelligents qui aident les utilisateurs en ligne à surmonter la surcharge d'informations en fournissant des recommandations personnalisées sur divers produits.

Nous avons présenté dans ce chapitre, une vue d'ensemble des principes de bases de deux nouveaux types de systèmes de recommandation intégrant l'information de confiance et sémantique afin d'améliorer les lacunes des CARS.

Revue de littérature sur les systèmes de recommandation sensibles au contexte basés sur la confiance et la sémantique

Sommaire

5.1	Introduction	62
5.2	Les systèmes de recommandation sensible au contexte .	62
5.3	Les systèmes de recommandation basés sur la confiance (TARS)	64
5.4	Les systèmes de recommandation sensibles au contexte basés sur la confiance	66
5.5	Les systèmes de recommandation basés sur la sémantique	67
5.6	Les systèmes de recommandation sensibles au contexte basés sur la sémantique	68
5.7	Synthèse et conclusion	69

5.1 Introduction

Dans le domaine de la recherche sur les systèmes de recommandation sensible au contexte, des études approfondies ont été proposées. Ces études sont axées soit sur les systèmes de recommandation basé sur la confiance (TARS), soit sur les systèmes basé sur la sémantique, soit les deux ensembles. De ce fait, nous nous concentrons dans ce chapitre sur des travaux qui intègrent des informations contextuelles en utilisant différentes approches de CARS (pré-filtrage contextuel, post-filtrage contextuel et modélisation contextuelle). Puis, ceux qui intègrent les informations de réseau social et de confiance entre les utilisateurs dans TARS. Ensuite, les travaux qui utilisent la technologie du web sémantique. Nous allons aussi explorer des travaux qui les combinent. Nous terminons ce chapitre par une synthèse justifiant nos contributions.

5.2 Les systèmes de recommandation sensible au contexte

Compte tenu de l'importance des systèmes de recommandation tenant compte du contexte (CARS), plusieurs travaux ont été proposés afin de les améliorer.

Les auteurs dans [Villegas et al., 2018] fournissent un état de l'art de CARS afin de comprendre comment exploiter et intégrer les informations de contexte pour une recommandation précise selon les différents paradigmes du CARS (pré-filtrage, post-filtrage et modèle contextuel).

Les auteurs de [Sassi et al., 2017] présentent un aperçu complet des systèmes sensibles au contexte dans un environnement mobile pour identifier les informations contextuelles utilisées dans CARS, afin d'esquisser les futures directions de recherche possibles. Ils classent les systèmes de recommandation sensibles au contexte en six catégories principales en fonction de la nature des facteurs contextuels utilisés dans l'approche de recommandation.

Parmi les travaux proposés, les auteurs [Zheng et al., 2014] citent l'approche de découpage contextuelle (Context-aware splitting approach (CASA)). Les algorithmes de découpage de contexte sont considérés comme les plus efficaces et les plus populaires parmi les autres algorithmes de recommandation contextuelle [Baltrunas and Ricci, 2009], [Zheng et al., 2014], [Baltrunas and Ricci, 2014] qui appartient à l'ap-

proche de préfiltrage contextuelle . Cette dernière traite l'information en amont et produit une matrice d'évaluation bidimensionnelle (2D), qui nécessite après juste à appliquer l'un des algorithmes de recommandation traditionnels. Trois types d'algorithmes ont été proposés dans CASA : Item Splitting [Baltrunas and Ricci, 2009], User splitting [Said et al., 2011] et UI splitting [Zheng et al., 2014]. Ces algorithmes considèrent qu'un item (ou/et utilisateur) évalué dans deux contextes différents, est considéré comme deux nouveaux items virtuels (ou/et utilisateurs) en utilisant des tests statistiques appelés : critères d'impureté [Baltrunas and Ricci, 2014], [Campos et al., 2014]. Cependant, malgré le succès de CASA, cette approche souffre de problème de rareté des données après le fractionnement.

De plus, généralement, une seule condition contextuelle est utilisée pour appliquer le découpage, appelé : découpage simple ; Il est également possible d'effectuer une division complexe en utilisant plusieurs conditions sur plusieurs dimensions de contexte. Cependant, il y a des coûts importants de traitement ainsi la matrice devient de plus en plus sparse [Baltrunas and Ricci, 2014]. Les auteurs de [Yang et al., 2015] ont utilisé un algorithme d'optimisation discrète de l'essaim de particules binaires (BPSO) à des fins d'optimisation discrète dans la recherche de la combinaison de conditions contextuelles optimale et ont déterminé le nombre de conditions contextuelles à sélectionner pour découpage d'utilisateur. Les travaux de [Zheng et al., 2016], évaluent l'utilité du facteur d'émotion contextuelle dans un système de recommandation sensible au contexte. L'un des types d'algorithmes contextuels utilisés pour évaluer leurs performances est la division contextuelle. Récemment, les auteurs [Al Jawarneh et al., 2020] améliorent le traitement d'exécution de l'algorithme de découpage d'item en utilisant deep learning.

Une étude comparative pour analyser les meilleures approches de CARS (pré-filtrage, post-filtrage et modélisation du contexte) faite par les auteurs de [Campos et al., 2013] , indique que la plus performante dépend de la tâche de recommandation et du domaine d'application.

Panniello et al. [Panniello et al., 2009] introduisent une étude comparative entre approche pré et post-filtrage. Pour sélectionner la meilleure approche, nous devons comparer la précision entre méthode sans contexte et le pré-filtrage. Si la méthode sans contexte surpasse le pré-filtrage, le mieux est d'utiliser la méthode de post-filtrage. En d'autres termes, être rapide et raisonnable (cas du pré-filtrage) ou cher

mais potentiellement meilleur (cas du post-filtrage).

Les auteurs ont proposé une autre technique de post-filtrage sensible au contexte [Lee et al., 2016] et ont mis en place la méthodologie de compensation de contexte. La compensation de contexte est calculée en calibrant les valeurs moyennes pour différents contextes et par la suite, des items sont suggérés à l'utilisateur. Les résultats prouvent que la méthode proposée excelle plusieurs approches de pointe. Dans [Taneja and Arora, 2017], les auteurs proposent une autre technique de post-filtrage contextuel basé sur les clusters qui sépare implicitement les utilisateurs en clusters sur la base de dimensions contextuelles, puis reclasse les items en fonction de la probabilité de pertinence dans un cluster. Otebolaku et al. [Otebolaku and Andrade, 2017] ont prédit les préférences des utilisateurs à l'aide de l'algorithme kNN basé sur la similitude entre les contextes des utilisateurs et ceux des autres utilisateurs. Dans [Zheng et al., 2013], les auteurs ont intégré la pondération des dimensions contextuelles dans l'algorithme basé sur FC c-à-d utilisant l'approche modélisation contextuelle.

5.3 Les systèmes de recommandation basés sur la confiance (TARS)

L'idée principale derrière la recommandation basée confiance est d'utiliser les informations de confiance pour améliorer la qualité de la recommandation et pour relever certains des défis des systèmes de recommandation traditionnels. Dans ce qui suit, nous allons passer en revue les contributions les plus significatives.

Les auteurs de [Sinha et al., 2001] démontrent que, étant donné le choix entre les recommandations d'amis de confiance et celles des systèmes de recommandation, en termes de qualité et d'utilité, les recommandations d'amis de confiance sont préférées, même si les recommandations données par les systèmes de recommandation ont un facteur de nouveauté. La propagation de confiance est très utile pour gérer les problèmes de rareté des données. Les chercheurs ont suggéré différentes manières [Guo et al., 2014a] [Yadav et al., 2018].

[Jamali and Ester, 2009] combinent la confiance explicite et implicite mais ignore la similitude entre les utilisateurs. De plus, dans [Jamali and Ester, 2010], les auteurs proposent le modèle SocialMF en utilisant la factorisation matricielle avec propaga-

tion de confiance pour la recommandation.

[Guo et al., 2015] proposent un modèle TrustSVD basé sur le modèle SVD++ [Koren, 2010] prenant en compte des informations explicites et implicites de la confiance entre utilisateurs et les évaluations des items.

[Moradi and Ahmadian, 2015] améliorent les TARS en fournissant une nouvelle méthode pour améliorer la précision de la prédiction basée sur une nouvelle mesure de fiabilité. Cette mesure utilise non seulement la matrice de notation, mais prend également en compte les déclarations de confiance.

[Azadjalal et al., 2017] propose une nouvelle méthode, qui se base sur les déclarations de confiance pour le calcul de prédiction. Ce calcul est basé sur le calcul de degré de confidentialité pour la construction du réseau de confiance d'un utilisateur. Ils utilisent ensuite les concepts Pareto de dominance pour éliminer les utilisateurs qui ne fournissent pas des informations utiles pour le calcul de prédiction.

[Deebak and Al-Turjman, 2020] utilisent le concept de confidentialité dans les réseaux de services cloud Big Data dans le système de recommandation basé sur la confiance et basé sur la communauté. Les auteurs ont proposé une méthode qui intègre les voisins de confiance dans les techniques de filtrage collaboratif, visant à éviter les évaluations malveillantes. Leur approche s'est basée aussi sur le calcul de degré de confidentialité qui prend en considération le nombre d'évaluations impliquées et des conflits entre les opinions positives et négatives (c'est-à-dire les évaluations). La nouvelle approche de similarité est introduite en intégrant mesure de confidentialité dans la corrélation de Pearson traditionnelle, le même principe utilisé par [Azadjalal et al., 2017].

Dans notre deuxième contribution nous avons utilisé aussi ce concept de confidentialité. Le but d'un côté est d'alléger le problème de rareté ainsi démarrage à froid et d'un autre côté, s'assurer de la fiabilité des notes estimés pour construction du réseau de confiance.

[Parvin et al., 2019] proposent un algorithme de recommandation basé sur la confiance appelé : Filtrage collaboratif basé sur la confiance utilisant ACO (Ant Colony Optimization). Les auteurs combinent les déclarations de confiance en tant que source d'informations riche avec des valeurs de notation pour calculer les valeurs de similarité entre les utilisateurs. Ils ont adopté l'idée de [Moradi and Ahmadian, 2015] mais étendu afin de construire un ensemble d'utilisateurs fortement corrélés à

l'utilisateur cible.

[Yadav et al., 2018] utilisent un algorithme de méta heuristique basé sur l'intelligence en essaim pour optimiser les poids des métriques de confiance après avoir analysé différentes métriques de confiance.

[Wang et al., 2019] utilise des modèles de réseau de neurone basés sur l'auto-encodeur pour améliorer la précision de TARS et réduire la rareté des données et les problèmes de démarrage à froid. Les valeurs de confiance explicites et implicites sont prises en compte.

Dans [Gohari et al., 2017] utilisent les informations sémantiques et regroupent les items en fonction de leurs similitudes sémantiques. Ce regroupement (Clustering) vise à éliminer le problème d'Intérêts multiples et contenu multiple connu sous le nom (MIMC (Multiple-Interests and Multiple-Content)). Sachant que les algorithmes de filtrage collaboratif utilisent les mêmes voisins pour prédire tous les items invisibles pour l'utilisateur actif lors du calcul de similarité. Donc , le clustering permet au moment de la prédiction d'utiliser seulement le cluster adéquat à l'item cible.

5.4 Les systèmes de recommandation sensibles au contexte basés sur la confiance

Nous allons utiliser l'acronyme **T-CARS** pour représenter les travaux intégrant Confiance dans les CARS. Les travaux ci-dessous ont tendance à utiliser à la fois des informations de contexte et de confiance pour fournir des recommandations plus pertinentes et des prédictions de notation plus précises.

Dans [Liu, 2013], l'auteur propose un modèle de recommandation sociale contextuelle basé sur un réseau de confiance, qui prend en compte à la fois les informations contextuelles et sociales . Dans son travail, il utilise les tests statistiques pour évaluer les contextes, et la marche aléatoire (Random walk) pour collecter les notes les plus pertinentes. La machine de factorisation est appliquée pour prédire les notes manquantes , tout en incorporant les informations contextuelles dans le modèle de machine de factorisation.

[Li et al., 2014] étend la factorisation matricielle contextuelle (CAMF) qui appartient à l'approche de modélisation contextuelle [Baltrunas et al., 2011] à TCMF :

Factorisation matricielle contextuelle basée sur la confiance pour le filtrage collaboratif. TCMF utilise à la fois les informations de contexte et de confiance dans la phase de prédiction (biais de l'utilisateur et biais d'item) et l'interaction utilisateur-item-contexte-confiance. Ces travaux ont été étendus dans [Li et al., 2016], qui prennent en compte l'influence de la confiance dynamique et la transitivité de la confiance sur les notations basées simultanément sur l'analyse des réseaux sociaux.

[Keikha et al., 2015] fournit un système de recommandation sur le réseau social et utilise la marche aléatoire (random walk) pour sélectionner les items les plus pertinents, et dont les préférences de l'utilisateur sont définies en fonction du contexte des items.

Récemment, [Sejwal and Abulaish, 2019] propose une approche hybride en incorporant à la fois la confiance et le contexte des utilisateurs. Pour cela, les auteurs utilisent un filtrage collaboratif où la similitude est calculée entre chaque paire d'utilisateurs en fonction de la confiance et des fonctionnalités basées sur le contexte.

5.5 Les systèmes de recommandation basés sur la sémantique

Comme déjà mentionné dans le premier chapitre, le filtrage collaboratif est l'une des techniques les plus prometteuses dans les systèmes de recommandation.

En tant que nouvelle direction, des recommandations basées sur la sémantique ont émergé qui traitent de l'information sémantique des items. De tels systèmes peuvent améliorer les performances du FC classique en permettant au système de recommandation de faire des inférences basées sur une source de connaissances supplémentaire. Un bref état de l'art sur des travaux exploitant l'information sémantique et les données liées est présenté dans ce qui suit pour démontrer leurs avantages à améliorer la qualité de la recommandation.

[Shambour and Lu, 2012] ont développé un SR qui améliore la précision du FC en utilisant les caractéristiques sémantiques des items et du réseau de confiance sociale de l'utilisateur. Leur technique de recommandation basée sur la fusion sémantique de confiance gère à la fois les problèmes de rareté et de démarrage à froid.

[Fridi and Benslimane, 2017] confirment que l'ajout d'informations sémantiques

à partir du LOD améliore l'efficacité du filtrage collaboratif traditionnel.

[Nilashi et al., 2018] ont présenté un SR hybride qui combine FC, ontologie et Singular Value Decomposition (SVD). L'ontologie de domaine et le SVD sont adoptés pour résoudre les problèmes de rareté et scalabilité, respectivement.

Pour analyser les profils des utilisateurs d'après leurs tags afin de trouver des ressources intéressantes et les recommander. Dans le processus de recommandation, ils mettent en évidence le problème de l'ambiguïté qui peut être réduit en tenant compte des similarités sociales calculées sur les folksonomies combinées avec les similarités entre ressources dans les LOD.

L'information sémantique peut être utilisée aussi pour l'analyse des données sociales et la prédiction des liens, à savoir le travail de [Ghennane, 2015] affirme que les technologies sémantiques peuvent assurer la structuration et l'homogénéité des données produites par les différents médias sociaux. Le Web social et le Web sémantique ne sont pas concurrents mais plutôt complémentaires. Le but est d'enrichir le profil de l'apprenant avec un ensemble d'intérêts générés à partir des hashtags sémantisés et ainsi de proposer des ressources pédagogiques en adéquation avec les intérêts de l'apprenant dans le domaine de e-learning.

Dans [Beldjoudi et al., 2016], les auteurs utilisent LOD pour améliorer la recommandation de ressources.

[Javed et al., 2021] révèle que le système de recommandation basé sur une ontologie, combiné à d'autres techniques de recommandation, est universellement utilisé pour recommander des ressources contextuelles.

5.6 Les systèmes de recommandation sensibles au contexte basés sur la sémantique

Nous allons utiliser l'acronyme **S-CARS** pour indiquer les travaux intégrant la sémantique dans les CARS.

Les auteurs [Codina et al., 2013], utilisent une nouvelle approche de pré-filtrage contextuel qui surmonte la limitation majeure du paradigme de pré-filtrage contextuel en tirant parti des similitudes sémantiques entre les situations contextuelles.

Toujours, dans le cadre d'amélioration des CARS, les auteurs [Codina and Cec-

caroni, 2011] ont visé à étendre les systèmes de recommandation en incorporant des informations contextuelles et sémantiques. Ils appliquent l'approche de pré-filtrage contextuel au filtrage basé contenu amélioré sémantiquement.

[Allahyari and Kochut, 2016] proposent un modèle de topic probabiliste qui intègre les préférences des utilisateurs, les descriptions des items et les informations contextuelles basées sur des principes solides. Les informations contextuelles sont représentées comme un sous-ensemble de caractéristiques représentant les items, car le contexte est acquis à partir des descriptions sémantiques des items obtenus via DBpedia.

Aussi, vu l'intérêt des CARS dans le domaine de recherche, [Sejwal and Abulaish, 2021] développent une ontologie de film sensible au contexte (context-aware movie ontology (CAMO)) à partir de LOD et de deux bases de données de films. CAMO contient des concepts de film, des relations et diverses caractéristiques contextuelles de représentation et d'interaction. Elle semble une base de connaissances cinématographique utile pour le développement de systèmes de recommandation de films sensibles au contexte, et plus particulièrement pour le développement de techniques de recommandation contextuelles basées sur des items.

un travail récent [Berkani et al., 2021], consiste à combiner les trois informations à savoir social, sémantique et contexte pour améliorer la qualité de la recommandation.

5.7 Synthèse et conclusion

Nous avons exploré l'ensemble des travaux au tour des SRs, et plus particulièrement CARS. Le point communs que vise ces travaux est en premier lieu l'amélioration de la qualité de prédiction, ainsi diminuer les problèmes de démarrage à froid et rareté.

De ce fait, d'un coté , les travaux autour de TARS ont montré une amélioration considérable par rapport aux systèmes de recommandation traditionnel en terme de précision.

Et d'un autre coté, nous avons vu que les technologies du web sémantique ont leur part pour améliorer les SRs.

Le tableau (5.1) résume les travaux existants discutés dans cette sous-section et

aborde les techniques, les avantages/inconvénients de chaque approche.

Tableau 5.1: Synthèse d'état de l'art

Référence	Type	Technologies	Avantage & inconvénient	Dataset
[Guo et al., 2014b]	TARS	<ul style="list-style-type: none"> • TCF basé user • Merging method 	+ Confiance explicite + propagation de confiance + Rating confiance ; - ignore contexte - confiance implicite	-Filmtrust -Epinion -Flixter
[Guo et al., 2014b]	TARS	<ul style="list-style-type: none"> • CF basé user • Trust antecedent framework 	+confiance implicite -confiance explicite - ignore contexte	-Epinion -CiaoDVD
[Guo et al., 2015]	TARS	<ul style="list-style-type: none"> • CF • Factorisation matricielle • weighted-Regularization technique 	+extend SVD++ +Confiance explicite et implicite -Influence trust - ignore contexte	-Filmtrust -Epinion -Flixter -CiaoDVD
[Azadjalal et al., 2017]	TARS	<ul style="list-style-type: none"> • CF • Merging method • Pareto dominance 	+Rating confidence +trust propagation -ignore contexte -similarité -ignore sémantique	-Filmtrust -Epinion
[Moradi and Ahmadian, 2015]	TARS	<ul style="list-style-type: none"> • CF • Reliability method 	+ confiance explicite + confiance implicite + fiabilité - trust propagation - ignore contexte - ignore sémantique	-Epinion -Flixter
[Jamali and Ester, 2010]	TARS	<ul style="list-style-type: none"> • CF • Random walk 	+confiance explicite -confiance implicite -ignore contexte -ignore semantic	-Epinion

Tableau 5.1 Synthèse d'état de l'art – Suite

Référence	Type	Technologies	Avantage & inconvénient	Dataset
[Deebak and Al-Turjman, 2020]	TARS	<ul style="list-style-type: none"> • CF • Internet of thing • Clustering 	+Rating confiance +confiance +problème d'attaque -trust propa- gation -Ignore contexte	-FilmTrust -Epinions - CiaoDVD.
[Parvin et al., 2019]	TARS	<ul style="list-style-type: none"> • CF • Méthode d'optimisa- tion des colonies de fourmis (ACO) 	+mesure de similarité +confiance + confiance implicite -trust propagation -rating confiance -sémantique	-FilmTrust -Epinions - CiaoDVD
[Gohari et al., 2017]	TARS	<ul style="list-style-type: none"> • CF • Méthode d'optimisa- tion des colonies de fourmis (ACO) • similarité sémantique 	+mesure de similarité +confiance + confiance implicite +sémantique +MIMC -trust propagation -rating confiance -ignore contexte	-MovieLens 100K - MovieLens 1M
[Yadav et al., 2018]	TARS	<ul style="list-style-type: none"> • CF • swarm intelligence • Bat algorithm • Particle swarm Opti- misation 	+Confiance implicite +mesure de similarité -Ignore contexte -confiance explicite -Propagation de confiance -sémantique	-MovieLens -FilmTrust -Epinions -CiaoDVD.
[Wang et al., 2019]	TARS	<ul style="list-style-type: none"> • CF • Denoising autoencoder 	+confiance explicite et impli- cite +similarité sémantique +sémantique -rating confiance -Propagation -ignore contexte	-Epinions -FilmTrust

Tableau 5.1 Synthèse d'état de l'art – Suite

Référence	Type	Technologies	Avantage & inconvénient	Dataset
[Liu, 2013]	T-CARS	<ul style="list-style-type: none"> • CF • Factorisation matricielle • Random walk 	+ confiance implicite + contexte -sémantique -trust propagation - Rating confidence	-Douban
[Li et al., 2014]	T-CARS	<ul style="list-style-type: none"> • CF • Factorisation matricielle 	+améliore CAMF +confiance implicite +trust propagation +contexte -confiance explicite -sémantique -pas RMSE	-LDOS- CoMoDa.
[Keikha et al., 2015]	T-CARS	<ul style="list-style-type: none"> • CF • Random walk 	+ confiance implicite + contexte -sémantique -confiance explicite - trust propagation -Rating confi- dence	-Epinions
[Xu et al., 2017]	T-CARS	<ul style="list-style-type: none"> • CF • Similarité contexte 	+mesure de similarité -similarité contexte -trust propagation -sémantique	-
[Sejwal and Abulaish, 2019]	T-CARS	<ul style="list-style-type: none"> • CF • Weight context 	+problème d'attaque +Mesure de similarité +trust propagation - sémantique -rating confidence	Construit à partir de Rot- ten Tomatoes and IMDB
[Shambour and Lu, 2012]	SARS	<ul style="list-style-type: none"> • CF basé Item • Similarité sémantique 	+sémantique + confiance implicite et explicite +problème attaque - contexte	-MovieLens -The Yahoo! Webscope R4

Tableau 5.1 Synthèse d'état de l'art – Suite

Référence	Type	Technologies	Avantage & inconvénient	Dataset
[Nilashi et al., 2018]	SARS	<ul style="list-style-type: none"> • CF basé user et item • SVD • Clustering 	+sémantique +Mesure de similarité -confiance implicite et explicite -contexte	-MovieLens -The Yahoo! Webscope R4
[Beldjoudi et al., 2016]	SARS	<ul style="list-style-type: none"> • CF basé utilisateur • LOD 	+sémantique +lien social -contexte -confiance	-del.ici.us
[Codina et al., 2013]	S-CARS	<ul style="list-style-type: none"> • CBF (basé contenu) • Vector Space Model (VSM) 	+sémantique +contexte +sémantique -confiance -RMSE	-Music - Tourism - MovieAdom -MovieLens -LibraryThing
[Berkani et al., 2021]	S-CARS	<ul style="list-style-type: none"> • CF basé utilisateur • CF bas • KNN 	+sémantique +lien social +contexte +confiance -MIMC -propagation	-Yelp -Epinions
[Codina and Ceccaroni, 2011]	S-CARS	<ul style="list-style-type: none"> • CBF • Ontologie 	+sémantique +contexte -confiance -MIMC -propagation	-MovieLens
[Allahyari and Kochut, 2016]	S-CARS	<ul style="list-style-type: none"> • CF • Ontologie • LOD • probabilistic topic model 	+sémantique +contexte -confiance -MIMC -propagation	-MovieLens 1M

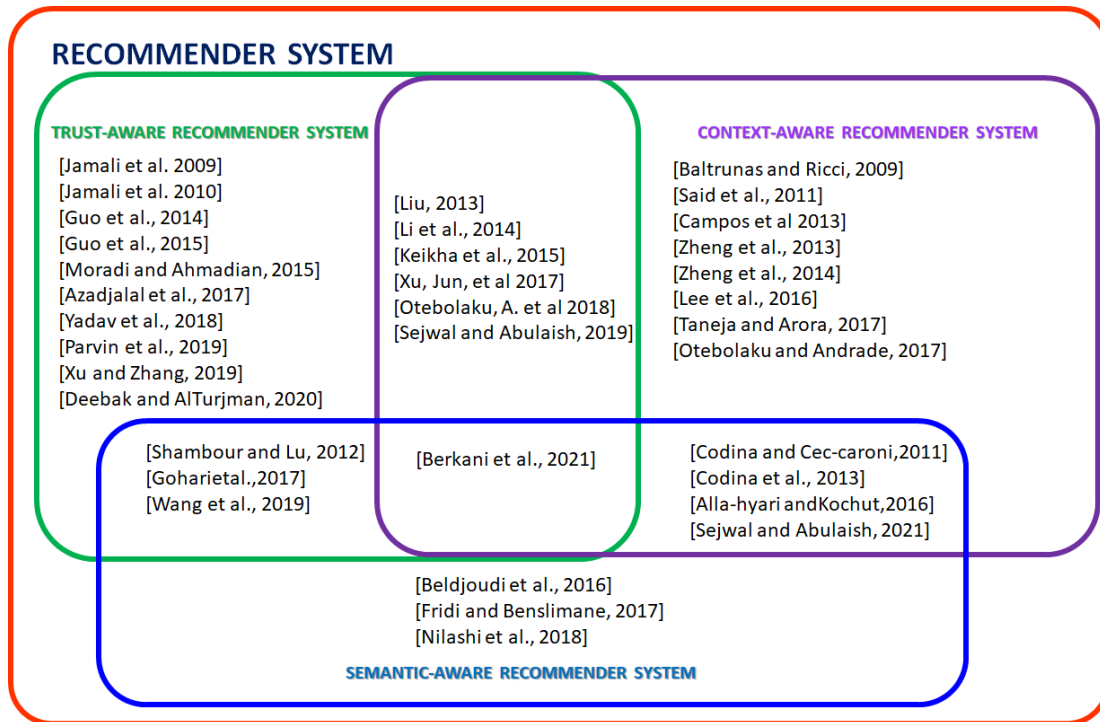


Figure 5.1: Synthèse des travaux connexes

Tableau 5.1 Synthèse d'état de l'art – Suite

Référence	Type	Technologies	Avantage & inconvénient	Dataset
[Sejwal and Abulaish, 2021]	S-CARS	<ul style="list-style-type: none"> • CF • Ontologie • LOD • OWL Validator model 	+sémantique +contexte -confiance -MIMC -propagation	leurs dataset

D'une manière général, d'après notre état de l'art, nous avons conclu que les travaux qui combinent l'un des deux informations à savoir : confiance et/ou sémantique améliorent la qualité de prédiction. La figure 5.1 donne un aperçu plus clair des différents travaux de notre état de l'art.

Pour résumer, nos premières motivations se sont focalisées à comment intégrer la confiance dans les CARS. Parmi les travaux sur les CARS, nous avons identifié que l'approche de découpage (Context Aware Splitting Approach (CASA)) appartenant à l'approche de pré filtrage contextuelle. Cette dernière était parmi les approches les plus efficaces et populaires qui améliore la recommandation contextuelle (résultat

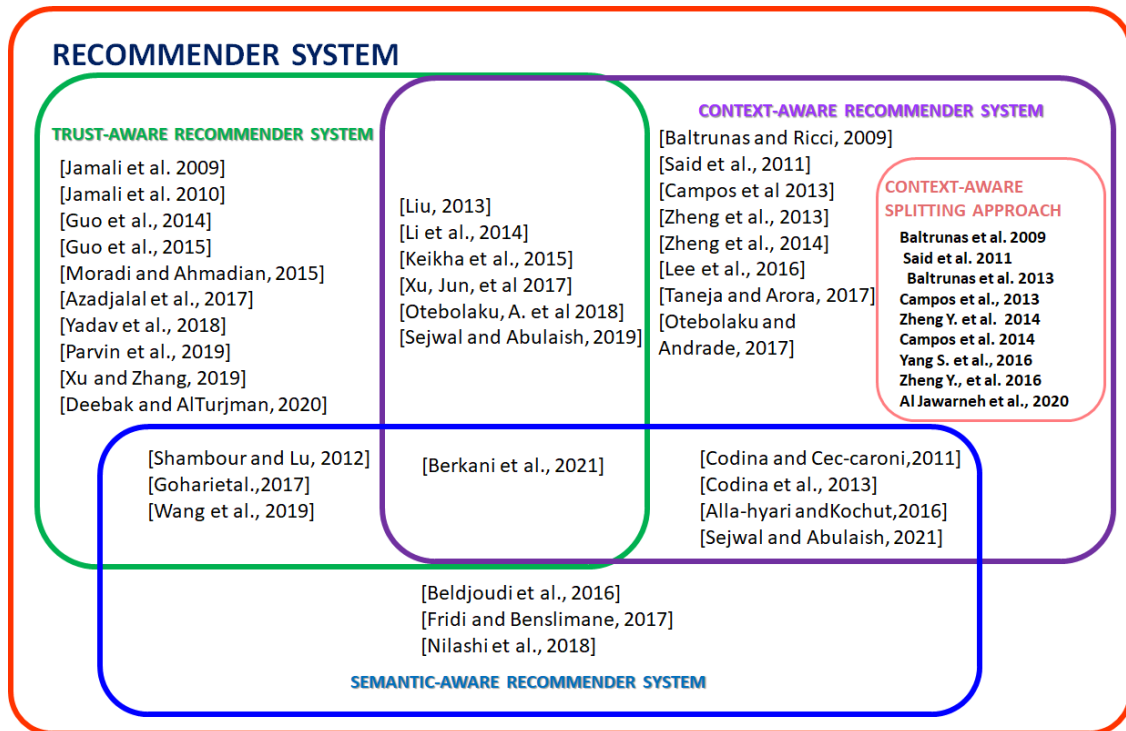


Figure 5.2: État de l'art de l'approche de découpage contextuel

de comparaison de [Campos et al., 2013]). Des travaux de recherche ont été explorés pour voir principalement l'apport de confiance sur cette approche. La figure 5.2 présente un aperçu sur l'état de l'art de ces nouvelles approches et chacune d'elles s'est avérées plus efficace que les systèmes de recommandation 2D traditionnels (càd pas de contexte ni d'informations de confiance). D'une part, les approches de découpage contextuel (CASA) améliorent la précision de prédiction des recommandations contextuelles mais néanmoins après le processus de fractionnement, la matrice de notation multidimensionnelle originale est transformée en une matrice bidimensionnelle qui souffre de rareté. D'autre part, nous avons constaté qu'une approche basée sur la confiance a l'avantage de réduire ces problèmes. Cependant, aucun travail antérieur n'évalue les algorithmes de découpage avec des modèles de recommandation tenant compte de la confiance.

La figure 5.2 présente une synthèse des travaux utilisant cette approche.

A partir de cette synthèse, nous avons proposé notre première contribution [el yebdri et al., 2019].

Ensuite, nous avons constaté que l'algorithme est très coûteux, comporte trois

boucles imbriquées coûteuses et doit vérifier de manière exhaustive toutes les valeurs de conditions de contexte pour chaque dimension de contexte de chaque item de la matrice de notation. De ce fait, nous avons ensuite étendu notre recherche, où nous avons constaté que les résultats des travaux de [Lee et al., 2016] qui utilise la méthode de compensation de contexte surpasse l'approche de découpage.

De ce fait, nous avons réorienté notre travail afin d'exploiter la confiance en se basant sur la méthode de compensation [El Yebdri et al., 2021].

Suite aux résultats obtenus de nos deux contributions, nous avons noté d'autres difficultés, à savoir, le choix de l'algorithme 2D utilisé avant ou après filtrage contextuelle, peut influencer sur le résultat de prédiction. Cependant, à partir des travaux cités dans l'état de l'art, nous avons tiré les points suivant :

- Utiliser un clustering d'items permet de se baser seulement sur les notes des items qui sont similaires à l'item cible
- Pour améliorer le FC, et avoir des prédictions diversifiées, utiliser la similarité sémantique
- Utiliser LOD pour enrichir la description de l'item ou le profil d'utilisateur qui va améliorer les résultats du calcul de similarité sémantique.
- l'intégration du contexte est faite au moment du calcul de prédiction (pas de pré-filtrage et post-filtrage)

d'où notre motivation à proposer une approche qui prédit les notes manquantes pour les utilisateurs dans un contexte spécifique de manière efficace, en utilisant principalement le réseau de confiance et la matrice de notation et éventuellement les améliorer par des informations sémantiques.

Partie II :
Contributions

Approches proposées

Sommaire

6.1	Introduction	79
6.2	Approche de découpage du contexte basée sur le réseau de confiance (TBCASA)	80
6.2.1	Architecture de l'approche découpage du contexte basée sur le réseau de confiance (TBCASA)	80
6.2.2	Enrichissement des items et utilisateurs	82
6.2.3	Pré-filtrage des données contextuelles	82
6.2.4	Construction du réseau de confiance	84
6.2.5	Sélection des voisins de confiance	85
6.2.6	Filtrage collaboratif basé sur le découpage de contexte et la confiance	86
6.2.7	Recommandation	86
6.3	Approche de post-filtrage contextuelle utilisant un réseau de confiance	87
6.3.1	Architecture du système TCPoFA	88
6.3.2	Pré-compensation des données contextuelles(compensation des ratings)	88
6.3.3	Construction du réseau de confiance initial	90
6.3.4	Estimation des valeurs de notations inconnues	92
6.3.5	Degré de confidentialité des valeurs estimées	92
6.3.6	Reconstruction du réseau de confiance	93
6.3.7	Processus de prédiction	94
6.3.8	Post-filtrage contextuelle (compensation des ratings)	94
6.3.9	Algorithme de notre approche	94

6.4	Approche de modélisation contextuelle basée sur la sémantique et la confiance	94
6.4.1	Architecture de l’approche de modélisation contextuelle basée sur la sémantique et la confiance (ST-CAMA)	96
6.4.2	Création de cluster d’items	97
6.4.2.1	Enrichissement sémantique de description des items	97
6.4.2.2	Clustering des items	98
6.4.3	Sélection des voisins de confiance	98
6.4.3.1	Calcul de similarité des utilisateurs basé sur le contexte	98
6.4.3.2	Construction du réseau de confiance	101
6.4.4	Calcul de Prédiction des notes inconnues	102
6.5	Conclusion	102

6.1 Introduction

Tout d’abord, à travers la synthèse tirée de notre état de l’art, la considération des informations extraites du réseau de confiance des utilisateurs et la connaissance du domaine sémantique, peut améliorer une approche de recommandation basée sur le contexte. Sur la base de cette notion et suite à nos recherches dans le domaine de CARS, nous allons présenter via ce chapitre nos approches qui permettent d’améliorer l’existant. Nous rappelons que ces systèmes souffrent de quelques limites à savoir : démarrage à froid et rareté et éventuellement le problème de scalabilité. Les solutions proposées dans ce chapitre sont basées essentiellement sur la prise en considération de l’information de confiance qui permet nettement de pallier les problèmes cités précédemment, et éventuellement l’information sémantique. Nos contributions sont :

1. Approche de découpage du contexte basée sur le réseau de confiance (Trust Based Context Aware Splitting Approach (TBCASA)) [El Yebdri et al., 2021]
2. Approche de post-filtrage contextuelle utilisant un réseau de confiance (Trust based Context aware Post Filtering Approach (TCPoFA)) [el yebdri et al., 2019]
3. Approche de modélisation contextuelle basée sur la sémantique et la confiance (ST-CAMA :Semantic and Trust based Context-Aware Modeling Approach)

6.2 Approche de découpage du contexte basée sur le réseau de confiance (TBCASA)

Les systèmes de recommandation sensibles au contexte sont utilisés pour fournir des recommandations personnalisées en exploitant la situation contextuelle. Ils prennent en compte non seulement les préférences de l'utilisateur, mais également des informations supplémentaires pertinentes (contexte).

Notre approche consiste à intégrer les utilisateurs, les items et les informations contextuelles dans un cadre unifié.

Tout d'abord, nous tenons à rappeler que les approches de découpage contextuel (CASA) [Baltrunas and Ricci, 2014] améliorent la précision de prédiction des recommandations contextuelles mais néanmoins après le processus de découpage, la matrice de notation multidimensionnelle originale est transformée en une matrice bidimensionnelle qui souffre de rareté et démarrage à froid. Une approche basée sur la confiance a l'avantage de réduire ces problèmes. Cependant, aucun travail antérieur n'évalue ces algorithmes avec des modèles de recommandation tenant compte la confiance. Nous proposons d'étendre cette approche, en ajoutant une information supplémentaire afin de palier les limites citées précédemment. Le détail de cette approche est présenté dans les sous sections suivantes.

6.2.1 Architecture de l'approche découpage du contexte basée sur le réseau de confiance (TBCASA)

La figure 6.1 présente l'architecture globale de notre proposition. Notre approche tient compte de plusieurs facteurs pour fournir une recommandation personnalisée. Dans un premier temps, nous tentons d'enrichir la description sémantique des items et d'utilisateurs en collectant des données à partir des données ouvertes liées, Linked Open Data (LOD). Ensuite, nous appliquons les algorithmes de découpage contextuelle afin d'avoir une matrice 2D avec des données contextualisées. Une fois le processus de fractionnement terminé, nous allons combiner les informations de confiance avec filtrage collaboratif basé utilisateur enrichi sémantiquement pour le calcul de prédictions. Notre objectif est d'avoir une recommandation contextualisée et basée sur des utilisateurs digne de confiance.

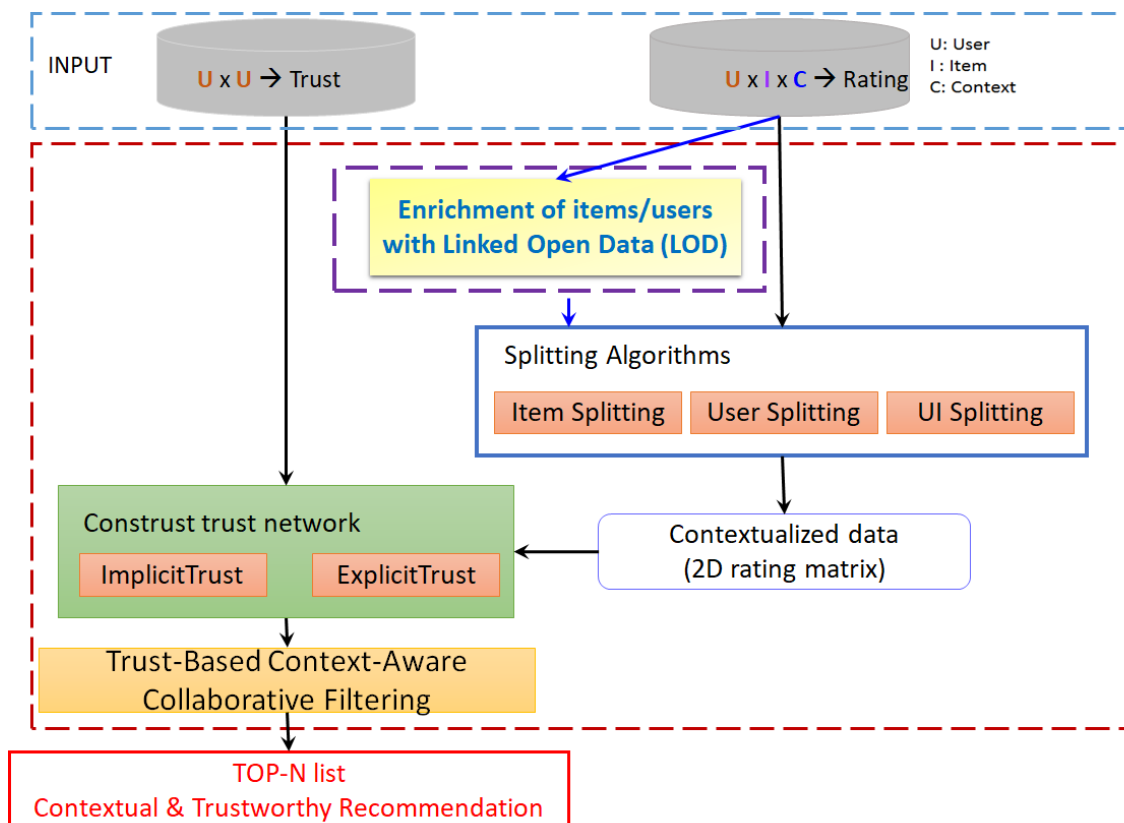


Figure 6.1: Architecture de l'approche découpage du contexte basé sur sémantique et contexte (TBCASA)

Nous détaillons dans ce qui suit les étapes de cette approche. Ainsi, dans le reste de cette proposition, nous nous intéressons uniquement à l'algorithme de découpage des items (item splitting).

6.2.2 Enrichissement des items et utilisateurs

Tout système de recommandation nécessite l'accès à une riche source d'informations, où le système obtiendra les informations et les filtrera en fonction de l'intérêt de l'utilisateur. Certaines recherches et applications ont utilisé DBpedia comme source d'information pour le système de recommandation. Notre travail s'appuie sur DBpedia pour récupérer le contenu.

L'objectif de cette étape est de valoriser la description des items et d'enrichir les profils des utilisateurs. Pour cela, nous proposons d'utiliser le Linked Open Data (LOD), considéré comme une source riche et fiable d'informations sur le contenu.

Après l'enrichissement d'un item i , il sera transformé en I' , où $I' = I + E$. par exemple :

$I = (12, \text{Jumanji}, (\text{année} : 1995), (\text{Genre1} : \text{enfants}), (\text{Genre2} : \text{aventure}), (\text{Genre3} : \text{fantaisie}))$). Après avoir enrichi les descriptions d'articles, I devient :

$I' = (12, \text{Jumanji}, (\text{année} : 1995), (\text{Genre1} : \text{enfants}), (\text{Genre2} : \text{aventure}), (\text{Genre3} : \text{fantasy}), (\text{Acteur1} : \text{Robin Williams}), (\text{Acteur2} : \text{Kirsten Dunst Diesel}), (\text{Writer1} : \text{Kirsten Dunst Diesel}))$.

Cette étape reste une proposition à explorer, non prise en considération dans l'expérimentation.

6.2.3 Pré-filtrage des données contextuelles

Nous avons choisi un algorithme de pré-filtrage contextuel, de ce fait, nous préparons nos données pour ensuite appliquer un algorithme de recommandation sur une matrice 2D. Parmi les algorithmes utilisées dans cette approche, nous nous intéressons à l'approche de splitting connu sous Context-Aware Splitting approach (CASA). Elle consiste à découper soit les items ou les users ou les deux de la matrice de notation en fonction du contexte.

En prenant en considération l'algorithme de découpage d'item, l'approche considère que deux items notés différemment dans deux conditions contextuelles différentes sont considérés comme deux items différents.

En d'autres termes, la nature d'un item, du point de vue de l'utilisateur, peut changer dans différentes conditions contextuelles, il peut donc être utile de le considérer comme deux items différents. Prenant l'exemple ci-dessous :

User	Item	Compagnon	Rating
U1	Frozen	Famille	5
U2	Frozen	Famille	5
U3	Frozen	Amis	2
U1	Frozen	Seule	3

Tableau 6.1: Matrice de notation avant l'algorithme du découpage d'item

À partir de cette matrice de notation (Tableau 6.1), nous remarquons que le film Frozen est bien noté quand le film est vu en famille, contrairement lorsqu'il a été vu seule ou avec amis. De ce fait, cet item sera subdivisé en deux items virtuels. Cette division n'est effectuée que s'il existe une preuve statistique selon laquelle, dans deux conditions contextuelles, les notes de l'item étaient différentes, c'est-à-dire que les utilisateurs évaluent différemment l'item. Le résultat de découpage de l'exemple précédent est représenté dans le tableau 6.2.

User	Item	Rating
U1	FrozenF	5
U2	FrozenF	5
U3	Frozen2	2
U1	Frozen2	3

Tableau 6.2: Matrice de notation après l'algorithme du découpage d'item

Nous divisons l'item en utilisant un découpage simple, ce qui signifie l'utilisation d'une seule condition contextuelle pour le découpage, car l'utilisation de plus d'une condition augmente la rareté des données. Plus précisément, pour chaque item, on itère chaque condition contextuelle de chaque dimension contextuelle. Si l'item i présente des différences significatives (en utilisant des critères d'impuretés [Campos et al., 2014]) dans la matrice de notation, nous créons deux nouveaux items artificiels i_c et $i_{\bar{c}}$, puis nous divisons le vecteur de sa notation en deux vecteurs, et chacun d'eux est affecté à un sous-ensemble des préférences des utilisateurs de l'item d'origine, en fonction de la valeur du contexte associée.

Pour les premiers résultats, nous utilisons `tmean` comme critère d'impureté. `tmean` compare deux échantillons et calcule la différence significative entre les moyennes des notes dans les deux sous-ensembles de notes. Plus la valeur `t` du test est grande, plus la différence des moyennes dans les deux échantillons est vraisemblablement significative. Nous n'obtenons comme résultat de cette étape que des données contextualisées et une matrice de notation bidimensionnelle.

Dans ce travail, nous nous concentrons sur une application simple de cette méthode où un item est découpé uniquement en deux items (on appelle ça : condition contextuelle binaire pour le découpage). Un découpage plus agressive d'un item en plusieurs items, pourrait produire encore plus d'items "spécialisés", mais potentiellement augmenter la rareté des données.

Cependant, comme indiqué précédemment, la matrice de notation résultante souffre de problèmes de rareté et de démarrage à froid. Pour résoudre ces problèmes, nous intégrons, à l'étape suivante, les relations de confiance sociale entre les utilisateurs pour calculer la prédiction.

6.2.4 Construction du réseau de confiance

Cette section exploite l'avantage de l'intégration du système de recommandation sensible à la confiance (TARS). De ce fait, afin de réduire le problème de démarrage à froid du CF traditionnel, nous pourrions construire le réseau de confiance des utilisateurs. Ces utilisateurs indiquent explicitement les relations de confiance. L'utilisateur actif u peut identifier un ensemble d'utilisateurs de confiance, désigné par T_u . Si l'utilisateur u fait confiance à l'utilisateur v , la valeur de la confiance est exprimée en $t_{u,v}$. C'est un nombre réel dans l'intervalle $[0, 1]$.

Nous construisons un réseau de confiance basé sur les informations de confiance explicites disponibles dans la matrice de confiance T . Nous exploitons un autre concept de réseau de confiance pour atténuer le problème de rareté des données dans la matrice de confiance en utilisant des algorithmes de propagation de confiance.

Pour cela, nous pouvons calculer la déclaration de confiance pour les utilisateurs où il n'y a pas de lien direct avec d'autres utilisateurs dans le réseau de confiance. Il est raisonnable de supposer que, si l'utilisateur u fait confiance à l'utilisateur b et b fait confiance à v , u peut faire confiance à v dans une certaine mesure.

L'équation 6.1 est utilisée pour calculer la déclaration de confiance entre les paires

d'utilisateurs comme [Massa and Avesani, 2007b] [Moradi and Ahmadian, 2015] :

$$t_{u,v} = \left(\frac{d_{max} - d_{u,v} + 1}{d_{max}} \right) \quad (6.1)$$

Avec :

$d_{u,v}$: la plus courte distance de propagation de la confiance entre utilisateur u et le confiant v

d_{max} : la distance de propagation maximale, qui représente une valeur entière positive. Par conséquent, les informations de confiance explicites sont généralement très rares, ce qui ne permet pas de prédire les évaluations si l'utilisateur n'a aucun ami de confiance. Cependant, intégrer également des informations de confiance implicites entre les utilisateurs reste un défi.

Pour cela, nous déduisons la relation de confiance implicite en utilisant une mesure de similarité pour évaluer la relation entre les utilisateurs. Ainsi, nous utilisons le Coefficient de Corrélation de Pearson (Equation 6.2) pour calculer la similarité entre deux utilisateurs comme suit :

$$sim_{u,v} = \frac{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_{u,v}} (r_{v,i} - \bar{r}_v)^2}} \quad (6.2)$$

Avec

$I_{u,v}$: est l'ensemble des items notés en communs entre l'utilisateur u et v ,

\bar{r}_u (\bar{r}_v) : est la moyenne des notations données par l'utilisateur u (v),

$r_{u,i}$: est la note donnée par l'utilisateur u à l'item i .

6.2.5 Sélection des voisins de confiance

Enfin, la valeur de poids finale ($w_{u,v}$) entre l'utilisateur cible et les autres utilisateurs est calculée en utilisant la fonction moyenne harmonique (Harmonic mean) de la similarité contextuelle incorporée et du score de confiance comme dans [Moradi and Ahmadian, 2015] pour obtenir des voisins de confiance similaires.

$$w_{u,v} = \begin{cases} \frac{2 \times sim_{u,v} \times t_{u,v}}{sim_{u,v} + t_{u,v}} & \text{if } sim_{u,v} \neq 0 \text{ and } t_{u,v} \neq 0 \\ t_{u,v} & \text{if } sim_{u,v} = 0 \text{ and } t_{u,v} \neq 0 \\ sim_{u,v} & \text{if } sim_{u,v} \neq 0 \text{ and } t_{u,v} = 0 \\ 0 & \text{if } sim_{u,v} = 0 \text{ and } t_{u,v} = 0 \end{cases} \quad (6.3)$$

Finalemnt, seuls les k premiers voisins de confiance similaires de l'utilisateur actif u (TN_u) sont sélectionnés pour calculer la prédiction.

6.2.6 Filtrage collaboratif basé sur le découpage de contexte et la confiance

Les utilisateurs voisins et digne de confiance seront utilisés pour prédire l'évaluation de l'utilisateur actif u pour l'item i sous un contexte cible c .

Pour cela, nous devons identifier la condition contextuelle cible de la note à prédire et l'un des nouveaux items correspondant avec les préférences générées sera utilisé pour la prédiction.

L'équation 6.4 est modifiée de tel sorte à prendre en considération non pas la similarité entre les pairs d'utilisateurs mais plutôt le poids (résultat de l'équation 6.3).

$$\check{r}_{u,i_c} = \bar{r}_u + \frac{\sum_{v \in TN_u} w_{u,v} (r_{v,i_c} - \bar{r}_v)}{\sum_{v \in TN_u} w_{u,v}} \quad (6.4)$$

Avec :

\check{r}_{u,i_c} : note prédite de l'item i pour l'utilisateur actif u par rapport au contexte cible

TN_u : les K voisins de confiance similaires à l'utilisateur u

r_{v,i_c} : note attribuée par l'utilisateur v à l'item i par rapport au contexte cible c

$w_{u,v}$: valeur de poids entre l'utilisateur u and v .

6.2.7 Recommandation

La phase de recommandation consiste à sélectionner les Top-N items comme liste de recommandations à suggérer à l'utilisateur actif.

6.3 Approche de post-filtrage contextuelle utilisant un réseau de confiance

Les systèmes de recommandations sensible au contexte sont un domaine d'étude actif car ils improvisent l'exactitude des recommandations. En raison de l'effet important des informations contextuelles dans les systèmes de recommandation, comme nous avons vu, une énorme quantité de travail a été mise en œuvre en utilisant l'une des trois approches, mais il y a encore place à l'amélioration.

Parmi ces travaux, nous nous intéressons à celui proposé dans [Lee et al., 2016], qui suit une approche contextuelle de post-filtrage. La particularité de cette méthode est qu'au lieu d'utiliser une mesure de similarité entre les contextes, elle calcule la compensation de chaque dimension contextuelle en mesurant leur moyenne et compense la notation pour appliquer après tout type d'algorithmes traditionnel de CF. Après cela, il ajuste le score de notation en tenant compte du contexte cible.

Cependant, ce dernier souffre de problèmes de démarrage à froid et de rareté car les utilisateurs évaluent une petite partie des items. Dans la réalité, les gens tendent beaucoup vers les goûts de leurs amis de confiance dans le réseau de confiance.

Trust-Aware Collaborative Filtering (TACF) améliore la prédiction des notes et résout les problèmes de filtrage collaboratif : démarrage à froid et rareté (parcimonie), car ils aident à connaître les préférences des utilisateurs dans les systèmes de recommandation.

Afin de résoudre les problèmes mentionnés ci-dessus, nous proposons TCPoFA (Trust based Context-aware Post Filtering Approach), une approche de post-filtrage sensible au contexte basée sur la confiance. Cette deuxième contribution, combine à la fois la confiance et les informations contextuelles, avant de calculer les recommandations, pour prédire une valeur de notation pour un item mesurant dans quelle mesure il convient au contexte cible.

Notre approche se compose de trois étapes principales : 1) Pré-traitement des données contextuelles en utilisant la méthode de compensation de contexte. 2) Compenser les notes connues et prédire les notes inconnues en se basant sur le réseau de confiance. 3) post-filtrage contextuel par décompensation des notes prédites et recommandation.

Dans la première étape, nous appliquons la méthode de compensation de contexte pour calibrer les valeurs moyennes des différentes dimensions contextuelles. Dans la deuxième étape, nous construisons le réseau de confiance initial de l'utilisateur actif

en utilisant une combinaison de valeurs de similarité et des déclarations de confiance pour prédire la notation initiale. Ensuite, pour chacune de ces évaluations, une valeur de confiance est calculée pour ne garder que les utilisateurs les plus importants dans le processus de recommandation. Enfin, la troisième étape consiste à récupérer le score de notation par rapport à la situation contextuelle de l'utilisateur actif.

6.3.1 Architecture du système TCPoFA

TCPoFA utilise les préférences des utilisateurs, réseau de confiance entre les utilisateurs, le contexte pour prédire les évaluations inconnues afin d'obtenir des recommandations contextualisées basées sur des utilisateurs dignes de confiance. Les principales étapes de l'approche proposée sont décrites dans la Figure. 6.2.

La première étape consiste à prétraiter les données contextuelles en utilisant la méthode de compensation de contexte. La deuxième étape estime les valeurs de notation inconnues à la base des notes et des déclarations de confiance en intégrant aussi les concepts de confiance (tenant compte de la fiabilité des notes initialement prédites). Enfin, la troisième étape donne une recommandation de filtrage post-contextuel. Nous détaillons ces étapes plus en détail dans les sous-sections suivantes.

6.3.2 Pré-compensation des données contextuelles(compensation des ratings)

Les informations contextuelles influencent les préférences de l'utilisateur. Dans notre travail, ces informations sont gérées en calculant la distance relative de chaque condition contextuelle. Cette étape consiste à appliquer une compensation pré-contextuelle pour ajuster les valeurs moyennes de différentes conditions contextuelles (voir Figure. 6.3).

En d'autres termes, nous calculons la note moyenne pour chaque condition contextuelle de chaque dimension contextuelle. Ensuite, nous définissons une valeur de compensation de chaque dimension contextuelle sous l'hypothèse suivante :

$$CompVal_{c_{dj}} = \begin{cases} 0 & \text{if } Avg_{c_{dj}} \geq Avg_{c_{dk}} \\ diff & \text{if } Avg_{c_{dj}} < Avg_{c_{dk}} \end{cases} \quad (6.5)$$

Avec :

$$diff = Avg_{c_{dk}} - Avg_{c_{dj}} \quad (6.6)$$

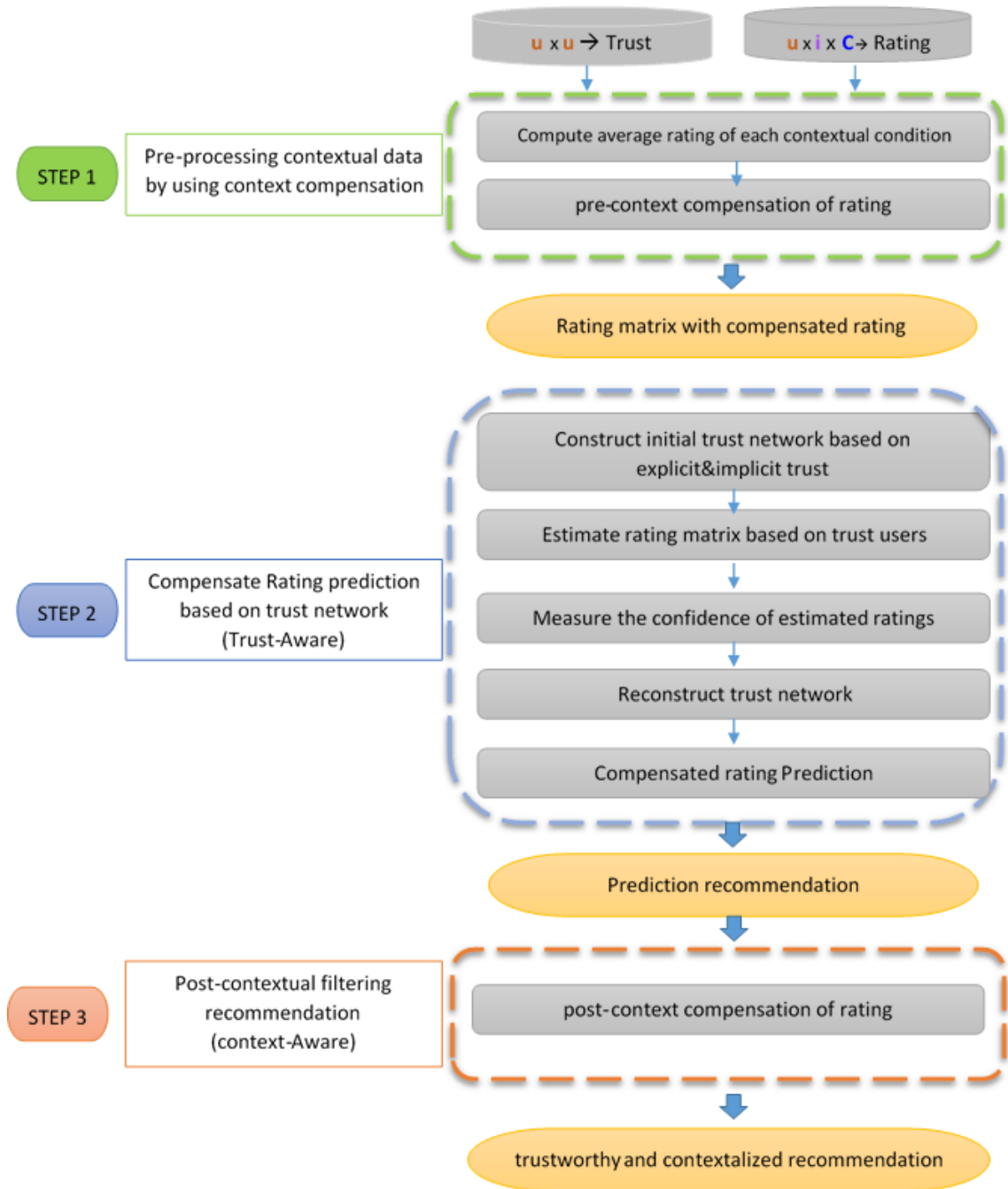


Figure 6.2: Présentation de l'approche de post-filtrage contextuelle basée sur la confiance (TCPoFA)

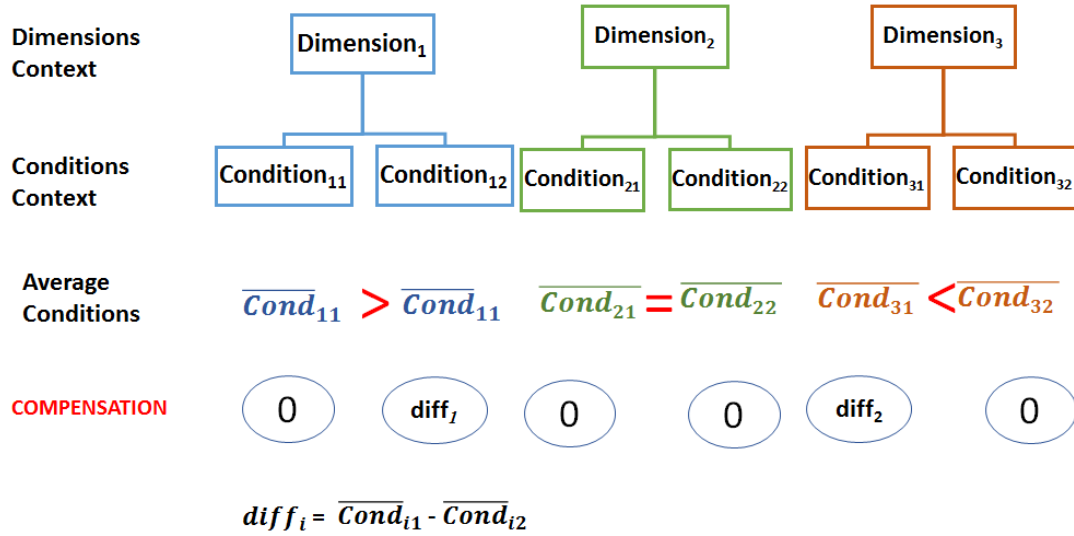


Figure 6.3: Compensation pré-contextuelle de la notation (pour une dimension contextuelle)

$CompVal_{c_{dj}}$: est la valeur à compenser pour une condition j de la dimension C_d

$Avg_{c_{dj}}$ ($Avg_{c_{dk}}$) : est la moyenne des notations de la condition contextuelle j (k) de la dimension contextuelle C_d .

Une fois l'ensemble des valeurs de compensation défini, nous équilibrons toutes les notes en fonction de la situation contextuelle de chaque tuple. Ainsi, toute note connue compensée r_{uiC} est exprimée par \ddot{r}_{uiC} :

$$\ddot{r}_{uiC} = r_{uiC} + \sum CompVal_c \quad (6.7)$$

Nous soustrairons cette somme à l'étape de post-filtrage.

6.3.3 Construction du réseau de confiance initial

Dans cette étape, nous considérons uniquement l'évaluation de la matrice 2D en utilisant uniquement utilisateur x item $\rightarrow \ddot{r}_{ui}$ (avec évaluation compensée). Cependant, la matrice de notation qui en résulte souffre de problème de rareté. Les informations de confiance jouent un rôle très important pour surmonter cette limitation. Pour cela, dans nos travaux, nous considérons également la relation de confiance entre les utilisateurs.

Nous construisons un réseau de confiance basé sur les informations de confiance explicites disponibles dans la matrice de confiance T. Nous rappelons que ce réseau est dirigé et pondéré. Les nœuds représentent les utilisateurs et l'arc représente les

déclarations de confiance disponibles entre les deux nœuds.

Nous exploitons un autre concept de réseau de confiance pour atténuer le problème de rareté des données dans la matrice de confiance en utilisant des algorithmes de propagation de la confiance. Pour cela, nous pouvons calculer la déclaration de confiance pour les utilisateurs où il n'y a pas de chemin direct avec d'autres utilisateurs dans le réseau de confiance. L'équation ?? (utilisée dans l'approche précédente) est utilisée pour calculer la déclaration de confiance entre une paire d'utilisateurs comme [Massa and Avesani, 2007b] [Moradi and Ahmadian, 2015].

La valeur de poids finale ($w_{u,v}$) entre l'utilisateur cible et les autres utilisateurs est calculée en utilisant la fonction moyenne harmonique (Harmonic mean) basée sur la similitude et la confiance des utilisateurs pour obtenir des voisins de confiance similaires.

$$w_{u,v} = \begin{cases} \frac{2 \times sim_{u,v} \times t_{u,v}}{sim_{u,v} + t_{u,v}} & \text{if } sim_{u,v} \neq 0 \text{ and } t_{u,v} \neq 0 \\ t_{u,v} & \text{if } sim_{u,v} = 0 \text{ and } t_{u,v} \neq 0 \\ sim_{u,v} & \text{if } sim_{u,v} \neq 0 \text{ and } t_{u,v} = 0 \\ 0 & \text{if } sim_{u,v} = 0 \text{ and } t_{u,v} = 0 \end{cases} \quad (6.8)$$

Nous utilisons le coefficient de corrélation de Pearson (Pearson correlation coefficient (PCC)) pour calculer la similarité ($sim_{u,v}$) entre deux utilisateurs comme suit :

$$sim_{u,v} = \frac{\sum_{i \in I_{u,v}} (\ddot{r}_{u,i} - \bar{\ddot{r}}_u)(\ddot{r}_{v,i} - \bar{\ddot{r}}_v)}{\sqrt{\sum_{i \in I_{u,v}} (\ddot{r}_{u,i} - \bar{\ddot{r}}_u)^2} \sqrt{\sum_{i \in I_{u,v}} (\ddot{r}_{v,i} - \bar{\ddot{r}}_v)^2}} \quad (6.9)$$

Avec

$I_{u,v}$: est l'ensemble des items notés en communs entre l'utilisateur u et v ,

$\bar{\ddot{r}}_u$ ($\bar{\ddot{r}}_v$) : est la moyenne des notations données par l'utilisateur u (v),

$\ddot{r}_{u,i}$: est la note donnée par l'utilisateur u à l'item i .

Seuls les premiers \mathbf{k} voisins de confiance similaires de l'utilisateur actif sont sélectionnés pour calculer la prédiction d'évaluation initiale. Ils sont également utilisés pour construire son réseau de confiance.

6.3.4 Estimation des valeurs de notations inconnues

Afin de réduire le problème du démarrage à froid et de la rareté des données, dans cette étape, nous estimons et attribuons des valeurs aux notes inconnues de la matrice de notation (voir Eq :6.10), qui sera utilisée dans les étapes suivantes. Pour cela, nous utilisons l'opinion des K voisins de confiance similaires sélectionnés précédemment .

$$\check{r}_{u,i} = \bar{r}_u + \frac{\sum_{v \in TN_u} w_{u,v} (\check{r}_{v,i} - \bar{r}_v)}{\sum_{v \in TN_u} w_{u,v}} \quad (6.10)$$

Avec :

$\check{r}_{u,i}$: note initiale attribuée,

TN_u : ensemble de voisins de confiance similaires de l'utilisateur u ayant noté l'item i ,

$\check{r}_{v,i}$: note attribuée par l'utilisateur v à l'item i

$w_{u,v}$: valeur de poids entre l'utilisateur u et v .

La matrice de notation est maintenant beaucoup plus complète que l'original.

Cependant, la qualité de ces notes estimées sont inconnues. Il s'agit de confidentialité des notes estimées.

6.3.5 Degré de confidentialité des valeurs estimées

L'équation 6.10 nous permet d'estimer certaines notations. Cependant, nous allons étudier la qualité de ces notations avant de les utiliser. Pour celà, nous utilisons une mesure de fiabilité pour évaluer la qualité des notes estimées. Cette mesure 6.10 nous aide à vérifier que les évaluations estimées sont correctes et fiables.

Deux facteurs susceptibles d'influencer les notations estimées doivent être pris en compte : le *nombre de notations impliquées* et les *conflits entre les opinions positives et négatives* parmi toutes les notations fusionnées. Plus précisément, la valeur fusionnée est susceptible d'être correcte et fiable lorsqu'un item reçoit plus d'évaluations de la part des voisins de confiance.

En revanche, un item qui reçoit quelques notes, donc la valeur fusionnée a tendance à être plus bruyante et peu fiable. À cette fin, la mesure suivante est utilisée [Guo et al., 2014a], [Deebak and Al-Turjman, 2020].

elle est calculée pour chaque note :

$$C_{u,i} = \frac{1}{2} \int_0^1 \left| \frac{x^{P_{u,i}} (1-x)^{N_{u,i}}}{\int_0^1 x^{P_{u,i}} (1-x)^{N_{u,i}} dx} - 1 \right| dx \quad (6.11)$$

Avec :

$C_{u,i} \in (0, 1]$ est la valeur de fiabilité de la note estimée $\check{r}_{u,i}$.

$P_{u,i}$ ($N_{u,i}$) sont le nombre d'opinions positives (j'aime) et négatives (n'aime pas) sur l'item i , respectivement, qui sont notées par les utilisateurs de confiance de l'utilisateur u , ($p_{u,i} = |r_{v,i}; r_{v,i} > r_{med}, v \in T_u|$ et $n_{u,i} = |r_{v,i}; r_{v,i} \leq r_{med}, v \in T_u|$)

\check{r}_{med} représente la valeur médiane, qui est déterminée à partir de la note minimale \check{r}_{min} jusqu'à la note maximale \check{r}_{max} dans la matrice de notation.

6.3.6 Reconstruction du réseau de confiance

L'objectif de cette étape est de supprimer les utilisateurs douteux du réseau de confiance de l'utilisateur. Nous calculons la similitude entre l'utilisateur actif et les utilisateurs de confiance en incorporant la valeur de confidentialité ($sim'_{u,v}$) en présence de la mesure de confidentialité. Pour cela, nous utilisons le coefficient de corrélation de Pearson basé sur la confiance (Confidence-based Pearson Correlation Coefficient (CPCC)), qui est largement utilisé pour calculer la similitude des utilisateurs [Azadjalal et al., 2017] [Deebak and Al-Turjman, 2020] [Ahmadian et al., 2020].

$$sim'_{u,v} = \frac{\sum_{i \in I_{u,v}} C_{u,i} (\check{r}_{u,i} - \bar{\check{r}}_u) C_{v,i} (\check{r}_{v,i} - \bar{\check{r}}_v)}{\sqrt{\sum_{i \in I_{u,v}} C_{u,i}^2 (\check{r}_{u,i} - \bar{\check{r}}_u)^2} \sqrt{\sum_{i \in I_{u,v}} C_{v,i}^2 (\check{r}_{v,i} - \bar{\check{r}}_v)^2}} \quad (6.12)$$

Avec :

$I_{u,v}$ est l'ensemble des items noté par u and v après l'estimation des ratings.

Après calcul de la similitude, seuls les utilisateurs dont la similitude est supérieure à un seuil prédéfini Θ_c sont retenus pour la construction du réseau de confiance de l'utilisateur actif u . $Tconf_u$ est l'ensemble des utilisateurs avec des valeurs de confiance plus élevées.

$$Tconf_u = \{v | sim'_{u,v} > \Theta_c, v \in U\} \quad (6.13)$$

À la fin, la valeur de confiance finale est calculée comme suit :

$$t'_{u,v} = sim'_{u,v} \times t_{u,v} \quad (6.14)$$

6.3.7 Processus de prédiction

Nous calculons une prédiction de notation compensée pour prédire les items non évalués de l'utilisateur cible. Nous utilisons l'équation 6.15 pour calculer la prédiction pour un item i pour l'utilisateur cible u mais avec une note compensée. Au lieu d'utiliser tout l'ensemble d'utilisateurs de confiance (I_u), nous utilisons juste un ensemble d'utilisateurs confiants de u ($TConf_u$).

$$\check{r}_{u,i} = \bar{r}_u + \frac{\sum_{v \in TConf_u} t'_{u,v} (\check{r}_{v,i} - \bar{r}_v)}{\sum_{v \in TConf_u} t'_{u,v}} \quad (6.15)$$

6.3.8 Post-filtrage contextuelle (compensation des ratings)

Après avoir généré un premier ensemble de recommandations, nous appliquons une approche de post-filtrage contextuelle. L'approche de post-filtrage contextuel ajuste les valeurs de prédiction obtenues en fonction de la situation contextuelle C_{sit} de l'utilisateur cible.

Pour chaque note prédite, nous calculons la note finale prévue en soustrayant la prédiction de notation compensée de l'ensemble des valeurs compensées. L'ensemble des valeurs compensées est calculé à travers la situation contextuelle de l'utilisateur cible.

$$\check{r}_{uiC_{sit}} = \check{r}_{u,i} - \sum CompVal_c \quad (6.16)$$

Avec :

r_{ui} est une prédiction de notation compensée

$\sum CompVal_c$ est l'ensemble des valeurs compensées par rapport à la situation contextuelle cible.

6.3.9 Algorithme de notre approche

Le processus global de prédiction est résumé dans L'algorithme 1

6.4 Approche de modélisation contextuelle basée sur la sémantique et la confiance

Nous présentons dans ce qui suit notre troisième approche, baptisée (ST-CAMA : Semantic and Trust based Context-Aware Modeling Approach). Comme son nom l'indique, elle se base sur l'amélioration de l'approche de modélisation contextuelle, en

Algorithme 1 Pseudo-code de l'approche post-filtrage contextuelles basée sur la confiance

Entrée(s): Matrice User-Item R, Matrice de confiance T, K : plus proche voisins de confiance de l'utilisateur actif, a : l'utilisateur actif, Θ_c : seuil pour sélectionner les utilisateurs confidents , d_{max} : la distance de propagation maximale $NDim$: Nombre de dimension $NCdt_c$: Nombre de condition de c

Sortie(s): la prédiction des notes inconnues

▷ Etape I : Compensation des ratings de R

pour d= 1.. $NDim$ **faire**
 pour cdt= 1.. Cdt_a **faire**
 Calculer la valeur compensée à l'aide de l'équation 6.5 et de l'équation 6.6
 fin pour
fin pour
Compensation des ratings (notes) dans R en utilisant l'équation. 6.7
 ▷ Etape II : prédiction de la note compensée basée sur le réseau de confiance
Calculer la déclaration de confiance entre a et les autres utilisateurs en utilisant l'équation. ??
Calculer les valeurs de similarité entre a et les autres utilisateurs en utilisant l'équation. 6.9
Calculer la valeur de poids en utilisant l'équation. 6.8
 TN_a = l'ensemble k des plus proches voisins de confiance de a en utilisant Eqs.(??,6.9,6.8)
pour chaque item non noté i in R **faire**
 Calculer la prédiction initiale $\check{r}_{a,i}$ en utilisant Eq.6.10
 Calculer la confiance (fiabilité ou confidentialité) de la note estimée $C_{a,i}$ en utilisant Eq.6.11
fin pour
pour chaque utilisateur $v \in TN_a$ de l'utilisateur actif a **faire**
 Calculer la similarité de confiance $sim'_{a,v}$ en utilisant Eq.6.12
 si $sim'_{a,v} < \Theta_c$ **alors**
 supprimer v du réseau de confiance (voir eq.6.13)
 fin si
fin pour
Calculer la valeur de confiance finale à l'aide de l'équation Eq.6.14
Calculer la valeur de notation compensée à l'aide de l'équation Eq.6.15
 ▷ Etape III : Post-filtrage contextuelle
Calculer la prédiction de notation à l'aide de l'équation Eq.6.16
end

tirant parti des avantages qu'offrent TARS et SARS.

Nous avons vu à partir de notre état de l'art que la majorité des travaux ciblent d'atténuer les problèmes de démarrage à froid et rareté. Notre approche ST-CAMA prend en considération ces limites. Toutefois, un autre problème spécifique aux algorithmes de voisinage (KNN) est négligé (peu de travaux qui le prend en considération) à savoir : Intérêts multiples et contenus multiples (MIMC (Multiple-Interests and Multiple-Content)). Ce dernier vise à sélectionner uniquement les voisins qui ont des intérêts communs à l'item cible pour l'utilisateur actif. Les sous sections suivantes donne un aperçu de cette approche.

6.4.1 Architecture de l'approche de modélisation contextuelle basée sur la sémantique et la confiance (ST-CAMA)

Dans cette section, nous décrivons notre approche ST-CAMA (Semantic and Trust based Context-Aware Modeling Approach). La figure 6.4 présente l'architecture globale de notre proposition. Notre approche tient compte de plusieurs facteurs pour fournir une recommandation personnalisée. Dans un premier temps, nous incluons l'enrichissement de la description sémantique des items en collectant des données à partir de données ouvertes liées (LOD) pour calculer leur similarité afin de construire des clusters d'items. Ensuite, nous incorporons les informations contextuelles et de confiance entre les utilisateurs pour prédire l'évaluation de l'utilisateur actif à l'item dans le contexte cible.

ST-CAMA contient quatre composants principaux. Le premier composant consiste à créer des clusters d'items. Ces clusters sont utiles lors du calcul de la valeur de similarité entre les utilisateurs afin de sélectionner seulement les items proches de l'item à prédire. Le second consiste à intégrer la similarité de contexte basée sur le poids pour calculer la similarité entre chaque paire d'utilisateurs en prenant en compte uniquement les clusters appartenant à l'item cible. Le troisième composant intègre la similarité de contexte basée sur le poids pour calculer la similarité entre chaque paire d'utilisateurs en prenant en compte uniquement les clusters appartenant à l'item cible. À la fin, notre approche ST-CAMA permet de prédire les notes inconnues tout en prenant en considération le contexte cible de l'utilisateur ainsi les avis des voisins digne de confiance par rapport à un cluster d'item spécifique.

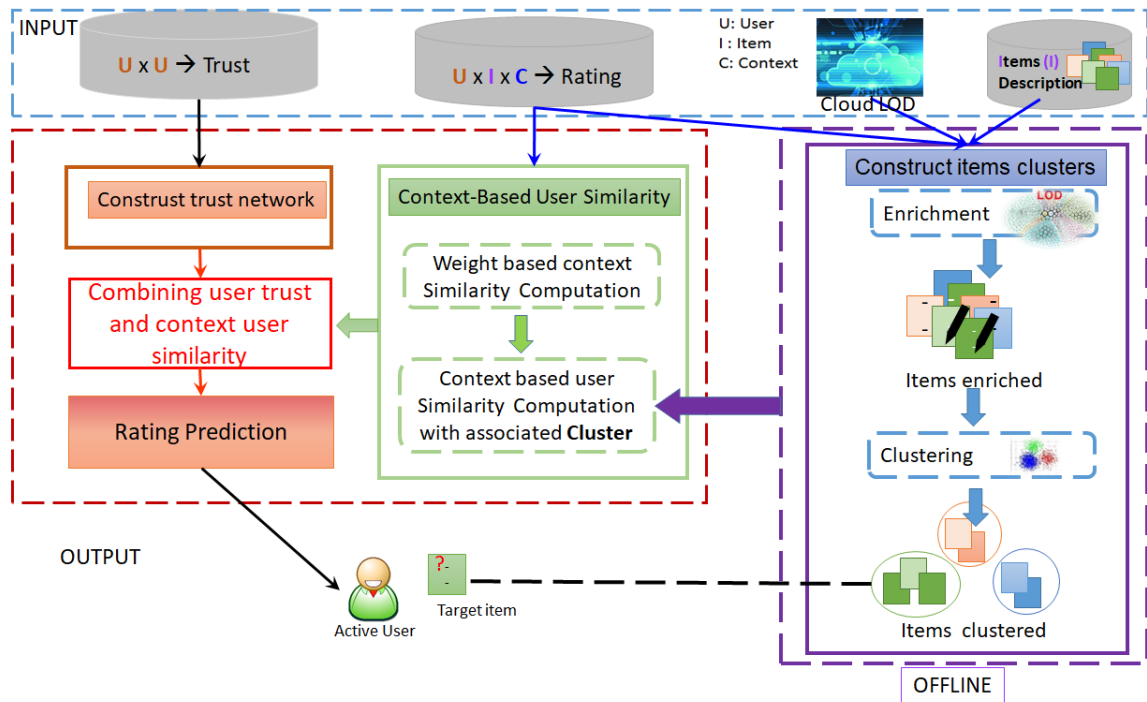


Figure 6.4: Architecture de l’approche de modélisation contextuelle basée sur la sémantique et la confiance (ST-CAMA)

6.4.2 Création de cluster d’items

Le but de cette étape est de regrouper les items en fonction de leurs similarité sémantique, afin de cibler lors de la sélection des voisins uniquement les items qui appartiennent au même cluster, ce qui signifie, prendre en considération l’avis des utilisateurs qui ont le même intérêt pour l’item cible. Elle s’exécute en mode hors ligne.

Pour cela, nous divisons cette étape en deux sous-étapes : **Enrichissement** et **clustering des items**.

6.4.2.1 Enrichissement sémantique de description des items

Chaque système de recommandation nécessite l’accès à une riche source d’informations, où le système récupérera les informations et les filtrera en fonction de l’intérêt de l’utilisateur. L’objectif de cette étape est d’enrichir sémantiquement la description des items . Un enrichissement sémantique consiste à ajouter des informations supplémentaire aux données de certaines ressources

Pour cela, nous proposons d’utiliser des données ouvertes liées (LOD), qui sont considérées comme une source riche et fiable d’informations sur le contenu. Certaines des recherches et applications utilisent DBpedia comme source d’informations pour le

système de recommandation.

Notre travail repose sur DBpedia pour récupérer le contenu et enrichir les descriptions des items. Par exemple dans un domaine de prédiction de film, grâce à DBpedia, on peut enrichir la description d'un film par d'autres propriétés.

6.4.2.2 Clustering des items

Dans cette étape, nous calculons la similarité entre deux items en fonction de leurs descriptions sémantiques enrichies. Nous utiliserons une mesure de similarité sémantique pour regrouper les items similaires.

Pour le clustering des items, nous allons adopter l'algorithme k-medoids en raison de sa simplicité et de sa grande précision. k-medoids est un algorithme de clustering basé sur des partitions. Nous choisissons le cluster qui sera utilisé au moment du calcul de la similarité de l'utilisateur actif avec d'autres utilisateurs dont le but est de se baser sur des items liés à l'item à prédire, pas sur tous les items (Problème de MIMC).

6.4.3 Sélection des voisins de confiance

L'intégration du contexte des utilisateurs et de la confiance dans le processus de recommandation est considérée l'une des approches prometteuses pour résoudre la plupart des problèmes rencontrés par les systèmes de recommandation traditionnels. De ce fait, d'un côté, nous allons chercher les utilisateurs similaires à l'utilisateur actif u , tout en prenant en considération le contexte cible et le cluster associé. Et d'un autre côté, nous allons calculer la confiance entre les utilisateurs pour construire le réseau de confiance de u .

De ce fait, la valeur de confiance et la similarité basée sur le contexte sont considérées pour le calcul de prédiction. Pour mieux expliquer la sélection des voisins de confiance, nous allons subdiviser cette partie en deux sous points : **calcul de similarité des utilisateurs basé sur contexte** et **calcul de confiance**.

6.4.3.1 Calcul de similarité des utilisateurs basé sur le contexte

Afin de prendre en considération les données contextuelles dans le calcul de similarité entre les utilisateurs, nous allons appliquer l'algorithme de kNN du filtrage collaborative basé utilisateur . De ce fait, il est nécessaire d'identifier l'ensemble des

utilisateurs (TN) qui ont des situations contextuelles similaires à la situation contextuelle de l'utilisateur actif :

$$TN(u) = v \in U | (Sim(Sit_u, Sit_v) > \theta_{cxt}).$$

Tel que :

$(Sim(Sit_u, Sit_d_v))$ est la similarité pondéré entre situation contextuelle de l'utilisateur active (Sit_u) et situation des autres utilisateurs (Sit_v).

A la base de cet ensemble (TN(u)), la corrélation de pearson est utilisée pour le calcul de similarité entre les utilisateurs.

1. Calcul de similarité entre les contextes

Pour calculer la similarité entre deux situations contextuelles des utilisateurs, généralement des similarités syntaxiques sont utilisées. Cependant, il est rare de trouver un utilisateur qui a noté un item dans une situation contextuelle identique , ce qui augmente le problème de rareté.

De ce fait, nous proposons dans notre travail d'utiliser une similarité sémantique pour pondérer les situations contextuelles, de sorte qu'il n'est plus nécessaire de trouver des valeurs de situations contextuelles exactement similaires.

Sachant que par exemple, dans un ensemble de données Film, la dimension «compagnon» a les valeurs : «amis», «famille», «seul». Il peut être judicieux de traiter les «amis» et la «famille» comme étant plus similaires que les «amis» et «seul» car ce sont des paramètres impliquant un groupe d'individus ; ce qui n'est pas possible de le définir ainsi si une mesure syntaxique est utilisée. En fait, pour le calcul de similarité de ces situations contextuelles : (weekend , family) avec (weekend , alone) et (weekend et children).

- Si nous utilisons par exemple une similarité de jaccard (6.19), nous allons avoir 1/2 et 1/2 pour les deux couples respectivement (seulement dimension Temps est similaire).

$$Sim_{jaccard}(Sit_u, Sit_v) = \frac{\sum_{C_d \in C_{sit} \cap C_{sitd}}}{\sum_{C_d \in C_{sit} \cup C_{sitd}} \quad (6.17)$$

Avec :

C_{sit} : situation contextuelle cible

C_{sitd} : situation contextuelle cible différente

$Sim_{jaccard}(sit, Sitd)$: étant le nombre total des conditions en commun/nombre total des dimensions

- Si nous utilisons une similarité sémantique, nous pouvons avoir un taux de ressemblance mieux entre les deux couples : (weekend , famille) et (weekend et enfant).

A cet effet, nous allons utiliser la mesure de similarité Wu and Palmer qui permet d'évaluer la proximité sémantique de deux concepts. Elle a l'avantage d'être simple à calculer en plus des performances qu'elle présente, tout en restant aussi expressive que les autres. Nous rappelons que cette mesure appartient à l'approche basé arc (Voir section 4.5.1.2). Le calcul de est fait en utilisant l'équation 6.18 :

$$Sim(Sit_{u_{ci}}, Sit_{v_{ci}}) = \frac{(2 * profondeur(PPS(Sit_{u_{ci}}, Sit_{v_{ci}})))}{(profondeur(Sit_{u_{ci}}) + profondeur(Sit_{v_{ci}}))} \quad (6.18)$$

Avec :

$PPS(Sit_{u_{ci}}, Sit_{v_{ci}})$ est le plus petit subsumant de $Sit_{u_{ci}}, Sit_{v_{ci}}$, i.e. c'est l'ascendant commun de $Sit_{u_{ci}}$ et $Sit_{v_{ci}}$ ayant la plus grande profondeur par rapport à la racine

$profondeur(Sit_{u_{ci}})$: est le nombre d'arcs qui séparent $Sit_{u_{ci}}$ et la racine de l'ontologie O (ontologie de domaine

$profondeur(Sit_{v_{ci}})$: est le nombre d'arcs qui séparent $Sit_{v_{ci}}$ et la racine de l'ontologie O.

Ce calcul est fait pour chaque dimension , d'où la valeur finale de pondération du contexte ($Sim(Sit_u, Sit_v)$) sera calculée comme suit :

$$Sim(Sit_u, Sit_v) = \frac{\sum_{i=1}^{C_i} Sim(Sit_{u_{ci}}, Sit_{v_{ci}}) * W_i}{Nb} \quad (6.19)$$

Avec :

$Sim(Sit_{u_{ci}}, Sit_{v_{ci}})$: similarité entre chaque dimension de la situation contextuelle

W_i : des poids donnés pour chaque dimension , sachant d'après les travaux [Zheng et al., 2013], [Zammali et al., 2015], [Dridi et al., 2020], pas toutes les dimensions influes sur la prédiction ainsi, les dimensions sélectionnées n'ont pas le même degré d'influence.

Enfin, la valeur de $Sim(Sit_{u_{ci}}, Sit_{v_{ci}})$ nous permet de contrôler la contribution du contexte lors de la prédiction des notes.

2. Calcul de similarité des utilisateurs basé sur le contexte

Nous allons calculer la similarité entre les pairs d'utilisateurs en utilisant le coefficient de corrélation de Pearson (PCC) (voir équation 6.20).

Toutefois, nous devons modifier cette dernière pour pouvoir incorporer la pondération des contextes ainsi prendre en considération le cluster associé à l'item cible.

$$sim_{(u,v)_{sitA}}^{C_i} = \frac{\sum_{i \in I_{u_a,v}^{C_i}} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v) Sim(Sit_{u_{ci}}, Sit_{v_{ci}})}{\sqrt{\sum_{i \in I_{u_a,v}^{C_i}} (r_{u,i} - \bar{r}_u)^2 \sum_{i \in I_{u_a,v}^{C_i}} (r_{v,i} - \bar{r}_v)^2 \sum_{i \in I_{u_a,v}^{C_i}} Sim(Sit_{u_{ci}}, Sit_{v_{ci}})^2}} \quad (6.20)$$

Avec

$I_{u_a,v}^{C_i}$: est l'ensemble des items notés en communs entre l'utilisateur u_a et v appartenant au cluster de l'item cible i

\bar{r}_{u_a} (\bar{r}_v) : est la moyenne des notations données par l'utilisateur u (v),

$r_{u,i}$: est la note donnée par l'utilisateur u à l'item i .

6.4.3.2 Construction du réseau de confiance

Nous exploitons toujours à travers notre travail, l'établissement de confiance explicite. Nous allons utiliser l'équation ?? utilisé dans les deux approches précédentes qui permet d'utiliser les valeurs de confiance qui sont directement fournies par les utilisateurs, tout en utilisant la propriété de transitivité pour propager la confiance.

Une fois nous avons défini les valeurs de confiance ainsi les valeurs de similarité entre les utilisateurs, nous allons utiliser la moyenne harmonique considérée parmi les mesures robustes pour le calcul du poids ajusté entre les utilisateurs u et v ; noté $W_{u,v}^{C_i}$. Ce calcul se fait en utilisant l'équation 6.8 mais modifiée de tel sorte que la similarité est celle calculée via l'équation 6.20.

Le résultat de cet étape est la constitution des voisins de confiance de l'utilisateur actif mais par rapport à un cluster spécifique. Par conséquent, le score de confiance et le score de similarité basé sur le contexte doivent être élevés pour obtenir une

pondération de similarité élevée, de ce fait, seuls les K premiers voisins de confiance similaires de l'utilisateur actif sont sélectionnés pour calculer la prédiction.

6.4.4 Calcul de Prédiction des notes inconnues

L'étape la plus importante dans un système de filtrage collaboratif est de produire la sortie en termes de prédiction. Comme déjà mentionné, la note prédite est obtenue grâce à un filtrage collaboratif basé utilisateur. Finalement, afin de prédire l'évaluation d'un item i par un utilisateur actif u , les utilisateurs top- k similaires à u sont identifiés à l'aide de l'équation 6.10.

6.5 Conclusion

Nous avons présenté nos contributions pour améliorer les systèmes CARS en utilisant des informations de confiance et enrichi par des technologies du web sémantique, le but est de diminuer tous les problèmes de rareté des données et de démarrage à froid.

Tout d'abord, nous avons intégré les informations de confiance dans l'approche de préfiltrage contextuelle à savoir : découpage d'item. cette approche est basée sur l'hypothèse que certains items peuvent avoir des évaluations différentes dans des contextes différents donc il faut les considérer deux items différents. Bien que la qualité de recommandation est améliorée face au système de recommandation traditionnel mais souffre toujours de problème de rareté. De ce fait, intégré les informations de confiance dans SR sont considérés un moyen efficace pour améliorer les performances de recommandation et améliore la rareté des données et les problèmes de démarrage à froid de CF.

Aussi, partant des résultats de [Lee et al., 2016] qui indique que l'utilisation de la méthode de compensation du contexte surpasse les approches de découpage, nous avons proposé notre deuxième contribution. Cette dernière a l'avantage de ne pas calculer la similarité entre les différentes informations contextuelles, contrairement au algorithme de découpage qui nécessite un traitement coûteux pour tester le découpage ou non de l'item. La méthode de compensation calibre les notes en fonction de la situation contextuelle de l'utilisateur actif et utilise l'un des algorithmes traditionnel de SR. Cependant, nous serons faces aux problèmes de rareté des données

et de démarrage à froid des algorithmes collaboratif pour faire la prédiction. De ce fait, nous avons proposé d'incorporer des voisins de confiance dans les techniques de filtrage collaboratif. Plus précisément, nous avons utilisé les évaluations des voisins de confiance pour compléter et représenter les préférences des utilisateurs actifs, sur la base desquelles des utilisateurs similaires. En plus que ça, nous avons mesuré la qualité des notations fusionnées par la mesure de confiance afin de garder que les avis des voisins digne de confiance. Les résultats d'expérimentation de ces deux approches sont détaillés dans le chapitre suivant.

Enfin, une autre approche en cours, tire les avantages de plusieurs travaux de recherche récents, c'est une approche hybride basée à la fois sur des informations de confiance et sémantique dans les CARS. Nous avons exploité la technologie du web sémantique afin d'enrichir et regrouper les items en clusters. Un cluster est sélectionné lors du calcul de similarité entre les différents utilisateurs, afin de prendre seulement en considération les notes attribués aux items similaires à l'item cible. aussi, le choix des voisins pour la prédiction ne se limite pas seulement aux préférences des utilisateurs mais aussi aux déclarations de confiances entre eux, afin d'avoir des recommandations digne de confiance.

Expérimentations

Sommaire

7.1	Introduction	104
7.2	Expérimentation du système TBCASA	105
7.2.1	Description de la base de Test	105
7.2.2	Résultats d'expérimentation	106
7.2.2.1	Paramètres de fractionnement	106
7.2.2.2	Résultats d'expérimentation	106
7.3	Expérimentation du système TCPoFA	108
7.3.1	Description de la base de Test	108
7.3.2	Résultats d'expérimentation	108
7.3.2.1	Impact de la propagation de la confiance	109
7.3.2.2	Impact de k (plus proche voisins de confiance)	109
7.3.2.3	Impact du seuil des utilisateurs fiables	110
7.3.2.4	Impact des informations de confiance	110
7.3.2.5	Comparaison avec les approches de l'état de l'art	111
7.4	Conclusion	114

7.1 Introduction

Nous nous sommes intéressés à la recommandation contextuelle améliorée principalement par les avantages que porte le web social, où les utilisateurs interagissent entre eux et expriment leurs avis les uns à l'égard des autres. Ces avis permettent aux SRs à base de confiance de remplacer et/ou compléter la similarité entre utilisateurs par des relations de confiance exprimées directement par les utilisateurs.

Dans ce chapitre, nous présentons la partie expérimentation des deux contributions étalées dans le chapitre précédent (6.2, 6.3). Nous testons nos contributions de recommandation en effectuant une expérience hors ligne. Les expériences hors ligne sont des expériences menées à l'aide d'ensembles de données ou de sources de données collectées à partir de l'interaction de l'utilisateur avec un système, mais le processus d'expérimentation n'interagit pas avec les utilisateurs. Nos expérimentations ont été faites sur un ensemble de données réel qui répond à nos exigences. Enfin, nous montrons les différentes expérimentations menées tout en discutant les résultats obtenus. Ces résultats sont comparés avec des algorithmes de recommandation contextuels et sensibles à la confiance de la littérature. Ces résultats révèlent que ces approches améliorent la pertinence des recommandations et surpassent les autres approches non contextuelles de la littérature en terme de précision.

7.2 Expérimentation du système TBCASA

Cette section montre des expériences pour analyser l'efficacité de notre approche 6.2. Les expériences sont effectuées sur un véritable jeu de données. En conséquence, nous présentons les résultats et comparons notre méthode avec des algorithmes de recommandation d'état de l'art.

7.2.1 Description de la base de Test

Dans notre travail, l'ensemble de données CiaoDVD [Guo et al., 2014b] du site *dvd.ciao.co.uk* est utilisé dans les expériences, dans lesquelles les utilisateurs évaluent divers items entre 1 et 5. De plus, ces utilisateurs peuvent également exprimer leurs déclarations de confiance avec les autres utilisateurs. Les valeurs des déclarations de confiance dans cet ensemble de données sont 0 ou 1. L'ensemble de données contient 17615 utilisateurs, 16121 films et 72521 évaluations. De plus, il existe 40133 relations de confiance entre 4658 utilisateurs. Nous avons enrichi cet ensemble de données avec différents concepts contextuels incluant le type de jour (weekday, weekend), les saisons (Autumn, Winter, Spring, Summer) pour évaluer notre approche.

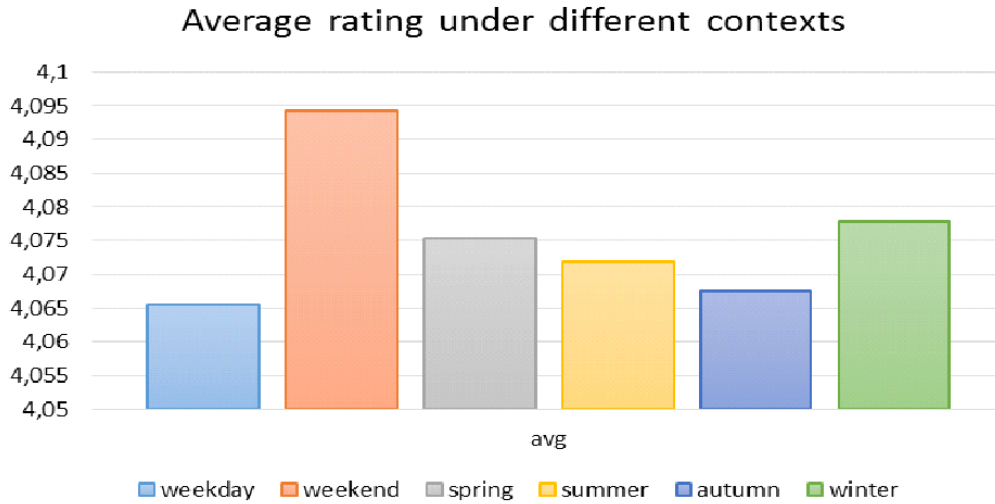


Figure 7.1: Note moyenne dans différents contextes dans l'ensemble de données CiaoDVD

7.2.2 Résultats d'expérimentation

7.2.2.1 Paramètres de fractionnement

Comme déjà mentionné, nous utilisons des critères d'impureté pour tester si un item nécessite d'être découpé. Nous utilisons dans notre travail tmeans [Baltrunas and Ricci, 2014] qui estime la signification statistique de la différence des moyennes des notations dans chaque condition contextuelle alternative, connu sous t-test :

$$T_{mean} = \left| \frac{\mu_{i_c} - \mu_{i_{\bar{c}}}}{\sqrt{\frac{\sigma_{i_c}^2}{n_{i_c}} + \frac{\sigma_{i_{\bar{c}}}^2}{n_{i_{\bar{c}}}}}} \right| \quad (7.1)$$

Avec :

μ_{i_c} : valeur d'évaluation moyenne de l'item i_c

$\sigma_{i_c}^2$: la variance de la valeur d'évaluation de l'item

n_{i_c} : le nombre de notes attribuées à l'item i

La figure 7.1 montre la valeur de notation moyenne calculée sur les différents contextes dans l'ensemble de données utilisé.

7.2.2.2 Résultats d'expérimentation

Dans cette section, un certain nombre d'expériences sont effectuées pour évaluer la qualité des prédictions de l'approche proposée. Nous utilisons la métrique MAE

Tableau 7.1: Principales approches de base utilisées dans notre expérimentation

Algorithm	UBCF	IKNN	TACF	Item splitting + UBCF	TBCASA
MAE	0,848	0,851	0,833	0,818	0,812

(Mean Absolute Error) qui permet de mesurer le degré auquel une prédiction est proche de la réalité.

CBCASA est comparé à plusieurs méthodes bien connues d'état de l'art, notamment :

1. le filtrage collaboratif basé sur l'utilisateur (UBCF)
2. le filtrage collaboratif basé sur l'item (IBCF)
3. le filtrage collaboratif basé confiance (TACF) qui tient en compte seulement la confiance
4. Item splitting (UBCF comme algorithme SR traditionnel)
5. notre approche TBCASA

Nous observons que notre approche TBCASA dépasse UBCF qui ne tient pas compte des informations de confiance et de contexte. De plus, l'approche de d'item splitting a obtenu de bon résultats par rapport à l'UBCF, tout en utilisant UBCF après le fractionnement des items, ce qui indique que les informations de contexte améliorent la prédiction mais que les informations de confiance ne sont pas incorporées.

La figure 7.2 représente le nombre d'items découpés pour chaque split (validation croisée 10 fois). Nous voyons à travers le graphique qu'environ 2,64% des items sont divisés, confirmant que les notes des utilisateurs sont influencées par des informations contextuelles.

Les résultats révèlent également que le TACF surpasse l'UBCF lorsque l'on considère uniquement l'influence des informations de confiance ignorant l'influence des informations contextuelles. Cela veut dire que l'intégration de l'information de confiance est utile pour améliorer les algorithmes classique de SR. Notre approche proposée, qui tient compte à la fois de la confiance et des informations contextuelles, améliore les recommandations pertinentes. Cette amélioration s'explique par l'addition de la probabilité que l'item soit noté par les voisins de l'utilisateur. De plus, l'opinion de ces autres utilisateurs est qualifiée de digne de confiance car elle est basée sur les déclarations de confiance.

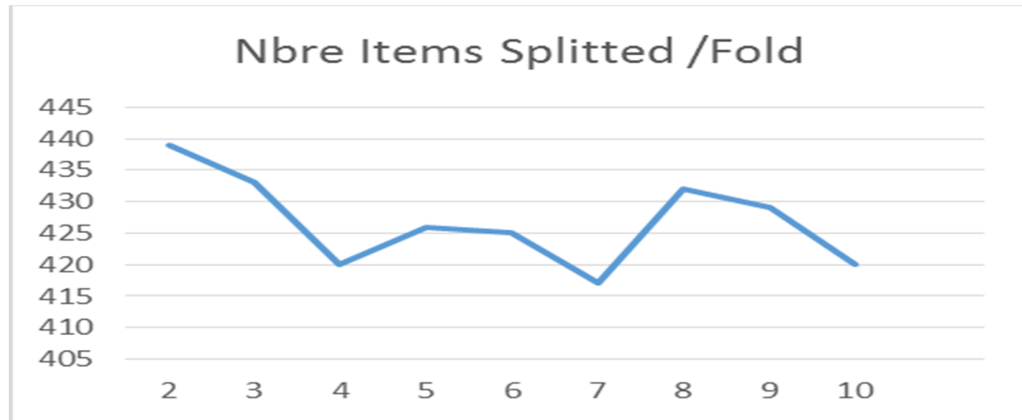


Figure 7.2: Note moyenne dans différents contextes

7.3 Expérimentation du système TCPoFA

Cette section présente les expériences d'évaluation de l'efficacité et de l'efficacité de TCPoFA sur un ensemble de données du monde réel. Ainsi, nous comparons notre approche avec des approches de pointe. Pour cela, nous décrivons d'abord cet ensemble de données et définissons les paramètres des expériences, puis nous discutons des résultats expérimentaux. Plus précisément, nous comparons la méthode proposée avec des travaux d'état de l'art (système de recommandation traditionnel, CARS, TARS).

7.3.1 Description de la base de Test

Nous allons utiliser le même jeu de données utilisé précédemment, à savoir CiaoDVDs. voir la section 7.3.1.

7.3.2 Résultats d'expérimentation

L'approche proposée utilise (couvre) plusieurs paramètres ajustables, dont D_{max} , k et Θ_c . Le réglage optimal de ces paramètres est déterminé par les expériences. La valeur du paramètre R_{med} est 4, ce qui représente la valeur médiane de toutes les notes de notre ensemble de données après l'étape de prétraitement des données contextuelles (Ajout de la valeur de compensation).

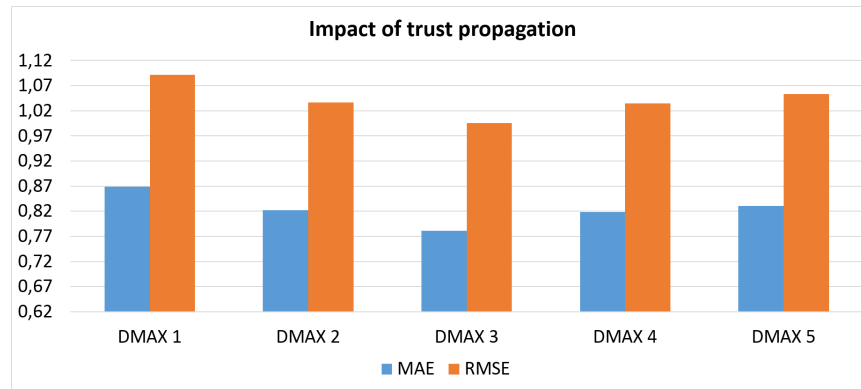


Figure 7.3: *Impact de taux de propagation de confiance*

7.3.2.1 Impact de la propagation de la confiance

Afin d'étudier l'impact de la propagation de la confiance sur la qualité de la recommandation, nous avons effectué des tests en modifiant la valeur de la distance de propagation de la confiance (D_{max}). Cependant, l'algorithme de propagation de confiance permet d'identifier plus de voisins de confiance, ce qui implique d'améliorer les performances de CF. La figure 7.3 montre qu'au début, les valeurs de MAE et RMSE augmentent, puis elle diminue lorsque l'on dépasse D_{max} 4.

Nous pouvons en conclure que la valeur de confiance implicite améliore la qualité de la recommandation mais en la limitant à une certaine valeur. D'après nos résultats (expériences), la distance de propagation de confiance courte $D_{max} = 3$ permet d'obtenir des performances satisfaisantes.

7.3.2.2 Impact de k (plus proche voisins de confiance)

Évaluons l'impact de k dans l'équation 6.9, qui définit le nombre de voisins de confiance les plus proches . Ce paramètre a aussi un impact important sur les performances de l'algorithme. Ainsi, il est nécessaire de déterminer la valeur optimale de ce paramètre. La figure 7.4 rapporte les résultats MAE et RMSE basés sur différentes valeurs du paramètre k , en utilisant $D_{max} = 3$.

À partir de la figure 7.4, on peut voir que lorsque la valeur k augmente le MAE et le RMSE diminuent, mais à $k = 60$ commence à augmenter. La valeur de la métrique est plus petite lorsque le paramètre k est de 50. Cela s'explique par le fait d'utiliser peu de voisins. Lorsque le nombre des voisins augmente qui veut dire prendre en considération dans la prédiction des voisins à faible niveau de similitude , conduit à

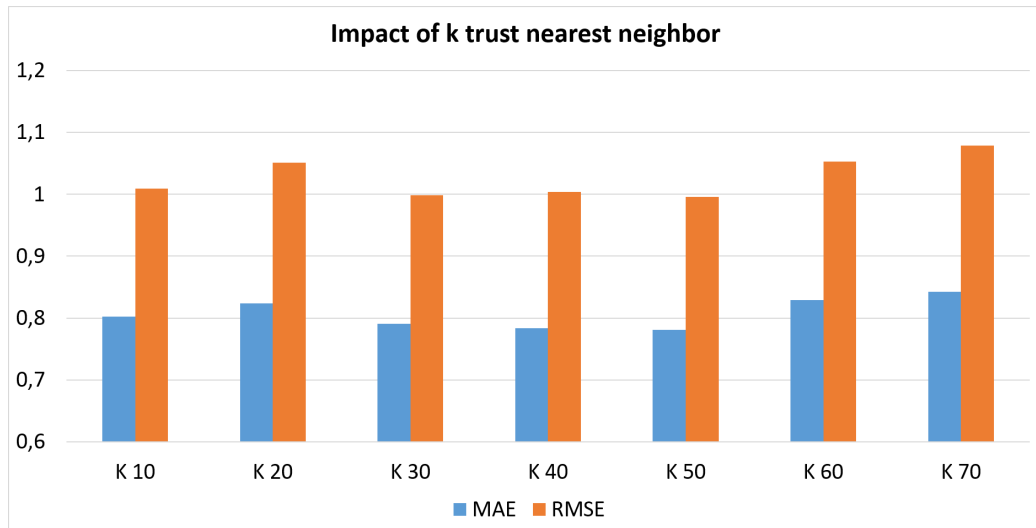


Figure 7.4: Impact de k (plus proche voisins de confiance)

une prédiction faible.

7.3.2.3 Impact du seuil des utilisateurs fiables

Θ_c est un autre paramètre ajustable qui est utilisé dans l'équation 6.13 pour identifier l'ensemble des utilisateurs de confiance de l'utilisateur actif. Ce paramètre élimine les utilisateurs moins fiables et sélectionne seulement les utilisateurs dignes de confiance pour l'utilisateur actif. Plusieurs expériences ont été menées pour analyser son impact. Les résultats montrent qu'au fur et à mesure que la valeur de confiance augmente, la valeur de MAE diminue, c'est logique, parce que le fait d'augmenter Θ_c , notre système TCPoFA écarte les utilisateurs méfiants, et seules les plus confiantes sont conservées pour générer la prédiction.

Néanmoins, avec $\Theta_c = 0,8$, on aura une bonne prédiction mais nous avons ajouté une autre métrique pour confirmer si ça reflète sur le nombre de notes prédit par le système, grâce à la métrique coverage (voir section :2.6.3).

Coverage permet de calculer le nombre de notes que l'approche est capable de prédire pour l'ensemble d'items disponibles. Nous confirmons à travers les résultats que certes, nous avons abouti à un bon MAE, mais ça reflète sur la possibilité de prédire des notes par notre système.

7.3.2.4 Impact des informations de confiance

Dans cette phase d'expérimentation, notre objectif est de démontrer que les concepts de confiance peuvent améliorer la prédiction dans les systèmes de recommandation.

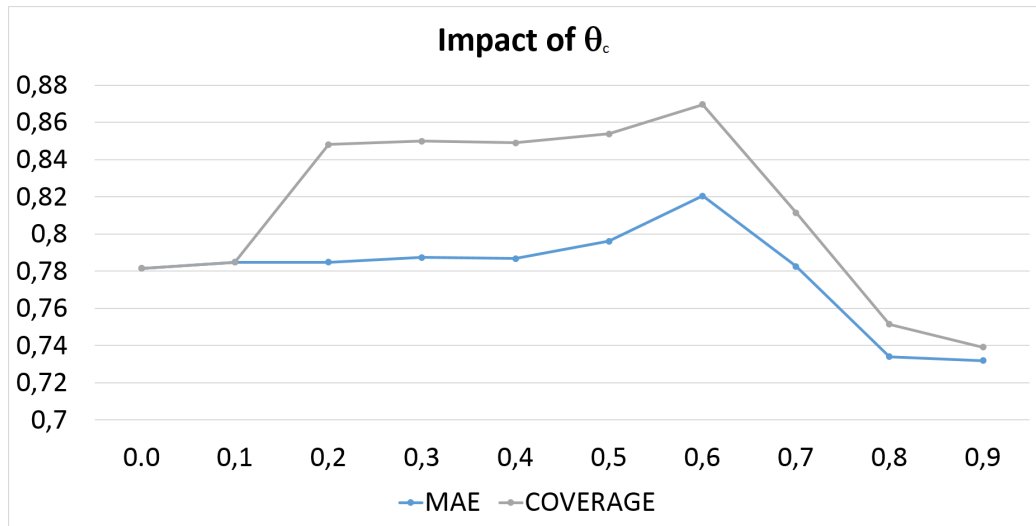


Figure 7.5: Impact du seuil des utilisateurs fiables Θ_c

A partir de la figure 7.6, nous notons que l'utilisation des informations de confiance améliore la recommandation et donne les meilleurs résultats. Plus précisément, lorsque nous incluons des valeurs de confiance comme mentionné dans la sous-section précédente.

7.3.2.5 Comparaison avec les approches de l'état de l'art

Nous avons comparé notre approche avec des algorithmes de recommandation sensible au contexte et confiance de la littérature. Nous utilisons dans nos expériences le moteur de recommandation open-source java basé sur le contexte nommé CARS-Kit [Zheng et al., 2015a], qui est spécialement conçu pour comparer des travaux de recherche sur les recommandations contextuelles.

De plus, nous utilisons une bibliothèque Java open source pour les systèmes de recommandation basé confiance, nommée LibRec [Guo et al., 2015], qui implémente une suite d'algorithmes de recommandation de pointe ainsi que les approches traditionnelles et celles qui intègrent des informations sociales, telles que TrustSVD.

Le tableau 7.2 résume les approches de pointe utilisées dans nos expériences.

Néanmoins, nous choisirons $K = 50$ dans nos prochaines comparaisons car il fournit le meilleur score de MAE et $D_{max} = 3$. Nous rendons compte de l'efficacité comparative de notre approche avec les approches de recommandation traditionnelles de

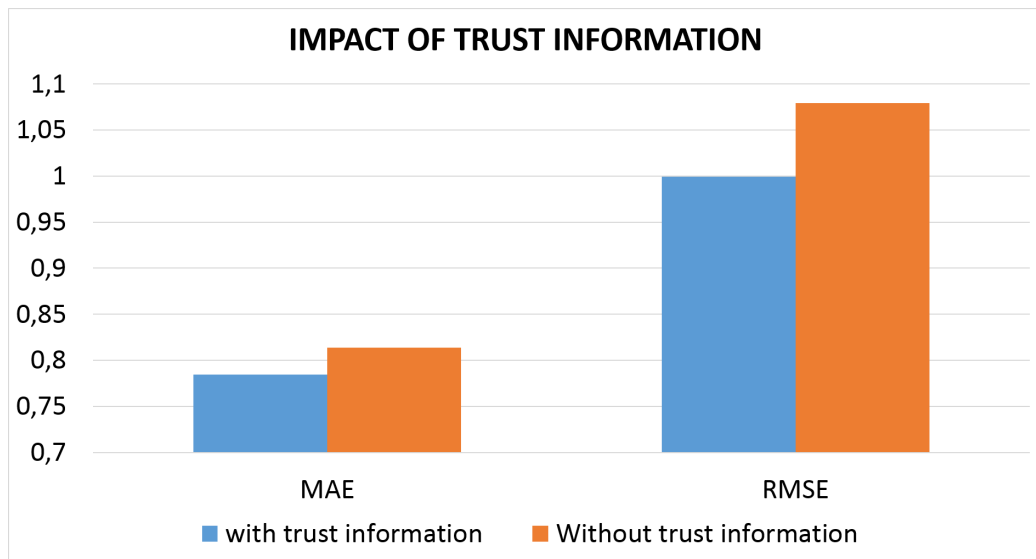


Figure 7.6: Impact des informations de confiance

Tableau 7.2: Principales approches de base utilisées dans notre expérience

Algorithme	Description	Categorie
User KNN	Filtrage collaboratif pur (CF) basé sur la similitude des évaluations des utilisateurs .	RS (LibRec)
SVD++ [Koren, 2010]	modèle Singular Value Decomposition : fait référence à un modèle de factorisation matricielle qui utilise les informations de l'historique des utilisateurs pour indiquer les préférences des utilisateurs	RS (LibRec)
UserSplitting-UserKNN [Said et al., 2011]	Un utilisateur évalue différemment dans deux conditions contextuelles différentes est considéré comme deux utilisateurs différents	CARS (CARSKit)
DCR [Zheng et al., 2012]	Le contexte est relaxé pour obtenir de meilleures performances de l'approche de modélisation contextuelle .	CARS (CARSKit)
TrustSVD [Guo et al., 2015]	Intègre à la fois les confiances explicites et implicites en tant que sources supplémentaires à SVD++	TARS (LibRec)

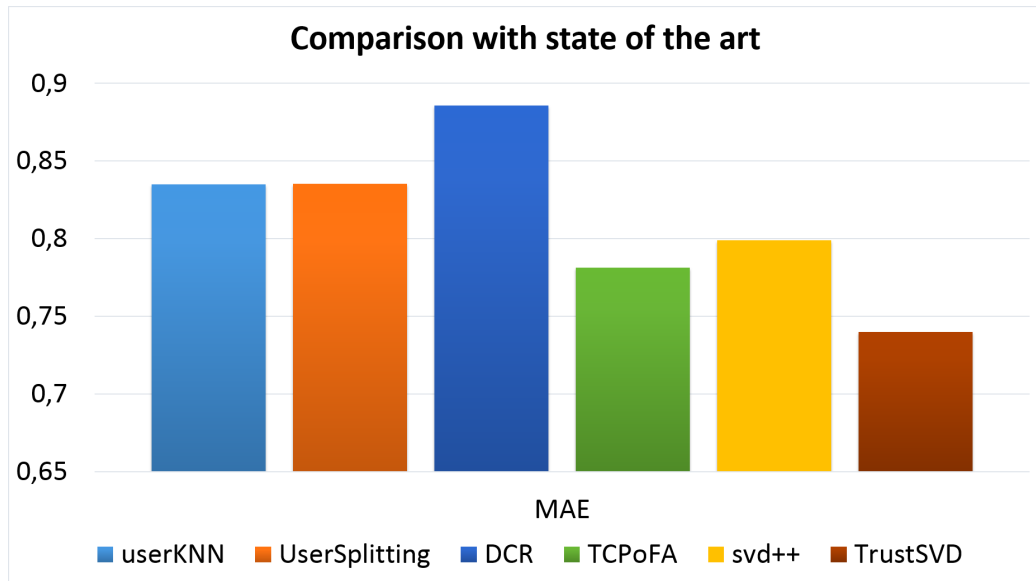


Figure 7.7: *Comparaison des résultats avec différentes approches de recommandation*

pointe, tenant compte du contexte et de la confiance. Les différentes valeurs de métriques des approches de recommandation considérées sont présentées dans la figure 7.7.

Nous observons (Figure 7.7) que UserSplitting et DCR (algorithmes CARS) surpassent UKNN et SVD++, ce qui indique que les informations de contexte améliorent la prédiction. TrustSVD surpasse également UKNN et SVD++ lorsque nous considérons uniquement l'influence des informations de confiance, ignorant l'influence des informations contextuelles, ce qui indique que les informations de confiance fournissent également une source (information) utile pour améliorer la prédiction de la recommandation.

Toutefois, les résultats indiquent clairement que l'approche proposée TCPoFA surpasse tous les algorithmes traditionnels (userKNN, SVD++) ainsi les algorithmes basés contexte (UserSplitting, DCR). Cela indique que la prise en considération de contexte et les déclarations de confiance pourraient fournir des informations utiles pour la recommandation. En ce qui concerne les résultats concernant TrustSVD avec TCPoFA, le résultat semble meilleur par rapport au nôtre. Nous supposons que l'utilisation de notre système avec le filtrage collaboratif basé modèle et plus précisément la factorisation matricielle peut donner de meilleurs résultats.

7.4 Conclusion

Nous avons présenté dans ce chapitre, l'ensemble des expérimentations de nos deux contributions.

Les résultats de ces expérimentations, nous ont permis de confirmer que d'un côté, l'information de confiance et contexte ont un impact positif sur la qualité de notes à prédire. et d'un autre côté, nous avons pu conclure que le choix de l'algorithme à utiliser dans les approches de pre-filtrage et post filtrage peut influencer sur la qualité de la prédiction (utilisation de KNN et non SVD++).

De ce fait, nous avons proposé une troisième approche qui se base sur une modélisation contextuelle , qui veut dire , que l'intégration du contexte est faite AU MOMENT de la prédiction ce qui ne nécessite ni un pré ni un post filtrage contextuel, ni utilisation d'algorithme de recommandation 2D. Cette approche est en cours de réalisation , et prend aussi en compte l'information sémantique pour voir son impact avec une telle approche hybride.

Conclusion et perspectives

Le domaine des systèmes de recommandation a été amélioré en passant des systèmes de recommandations classiques dits traditionnels aux systèmes de recommandations sensibles au contexte (CARS). La particularité de ces derniers est leur adaptabilité à l'environnement. Cela est fait par l'intégration de certaines informations dites contextuelles (exemple : le temps, l'emplacement... d'un utilisateur) pour effectuer une recommandation. Cependant, ces systèmes souffrent toujours de problèmes de démarrage à froid et rareté des données qui diminuent la précision des prédictions. Notre travail présente quelques améliorations par rapport aux approches de recommandations contextuelles existantes, en incorporant des informations supplémentaires. Principalement, nous avons exploité l'information de confiance et ce grâce à la disponibilité de relations sociales (par exemple, les réseaux sociaux) entre les utilisateurs (en plus de leurs historiques de notation sur les items). Les systèmes de recommandation sensible à la confiance constituent un vrai challenge pour aider les algorithmes de recommandation sensible au contexte à fournir des recommandations personnalisées plus appropriées. Aussi, les systèmes de recommandation basés sur la sémantique ont été utilisés avec succès dans différents domaines. De tels systèmes peuvent tirer parti du raisonnement sémantique pour fournir des recommandations beaucoup plus raisonnables dans le cas d'un item nouvellement ajouté ou dans des ensembles de données très rares.

8.1 Synthèse

La première partie de cette thèse est réservée aux travaux d'état de l'art, nous avons présenté les principaux concepts liés aux systèmes de recommandation et particulièrement les systèmes de recommandation sensible au contexte. Nous avons défini deux sous type de système de recommandation : système de recommandation sensible à la confiance et système de recommandation basé sur la sémantique, tout en mettant

en évidence aux concepts liées à ces derniers. Nous avons cloturé cette première partie par une revue de littérature sur les systèmes de recommandation sensibles au contexte basés sur la confiance et la sémantique afin de mieux positionner nos contributions.

La deuxième partie présente nos trois contributions. nous avons proposé et expérimenté une nouvelle approche qui suggère des items aux utilisateurs tout en prenant en considération le réseau de confiance des utilisateurs, le contexte des utilisateurs et la description des items. Pour traiter les informations contextuelles, nous avons amélioré les algorithmes de découpage, considérés parmi les plus efficaces pour une recommandation contextualisée. Ces algorithmes ne résolvent pas le problème de la rareté des données et du démarrage à froid. Pour cela, nous avons exploité les avantages du système de recommandation sensible à la confiance pour surmonter ces problèmes. Un réseau de confiance sera créé d'un utilisateur cible sur la base des déclarations de confiance entre les utilisateurs. Les résultats expérimentaux sur l'ensemble de données du monde réel ont indiqué des résultats satisfaisants. Puis nous avons proposé et validé une nouvelle approche de post-filtrage contextuelle basée sur la confiance. Plus spécifiquement, nous utilisons la méthode de compensation de contexte pour traiter les informations de contexte et nous utilisons le concept de propagation de confiance sur un réseau de confiance parmi les utilisateurs pour calculer les poids de similitude entre les utilisateurs. En outre, pour identifier les utilisateurs les plus dignes de confiance, des concepts de confiance sont utilisés. L'objectif est de proposer des items adaptés à la situation contextuelle et basés sur l'avis d'utilisateurs de confiance. L'approche proposée surpasse les autres approches non contextuelles et de base lorsqu'elle est évaluée par des mesures d'exactitude.

Et en dernier, une approche modélisation contextuelle basée sur la sémantique et la confiance pour sélectionner les voisins dignes de confiance de l'utilisateur actif en fonction de ses intérêts actuels pour un type spécifique d'items.

8.2 Perspectives

Plusieurs améliorations et extensions peuvent être envisagées pour enrichir les approches proposées :

- utilisation des informations de méfiance pour rendre notre approche plus précise.
- Appliquer des algorithmes basé modèle du filtrage collaboratif avant la phase de post-filtrage au lieu d'un algorithme basé mémoire

- Expérimenter l'apport des informations sémantiques sur les résultats obtenus. L'enrichissement des descriptions sémantiques des items et des profils d'utilisateurs peut également améliorer la qualité de notre approche proposée.
- Tester sur d'autres ensembles de données plus riches tels qu'Epinion.

Liste des publications

- Journaux internationaux avec comité de lecture

El Yebdri, Z., Benslimane, S.M., Lahfa, F. et al. Context-aware recommender system using trust network. Computing (2021). <https://doi.org/10.1007/s00607-020-00876-9>

- Conférences internationales avec comité de lecture

El Yebdri, Z., Benslimane, S.M., Lahfa, F. Trust and Context Aware Splitting Approach for Improving Prediction in Recommender System, Conference on Informatics and Applied Mathematics (IAM'2019). April 2019

Bibliographie

- [Abbasi et al., 2014] Abbasi, M. A., Tang, J., and Liu, H. (2014). Trust-aware recommender systems. *Machine Learning book on computational trust, Chapman & Hall/CRC Press*.
- [Abowd et al., 1999] Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., and Steggle, P. (1999). Towards a better understanding of context and context-awareness. In *International symposium on handheld and ubiquitous computing*, pages 304–307. Springer.
- [Aciar et al., 2007] Aciar, S., Zhang, D., Simoff, S., and Debenham, J. (2007). Informed recommender : Basing recommendations on consumer product reviews. *IEEE Intelligent systems*, 22(3) :39–47.
- [Adomavicius et al., 2005] Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A. (2005). Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)*, 23(1) :103–145.
- [Adomavicius and Tuzhilin, 2005] Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems : A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6) :734–749.
- [Adomavicius and Tuzhilin, 2011] Adomavicius, G. and Tuzhilin, A. (2011). Context-aware recommender systems. In *Recommender systems handbook*, pages 217–253. Springer.
- [Aggarwal et al., 2016] Aggarwal, C. C. et al. (2016). *Recommender systems*, volume 1. Springer.

- [Ahmadian et al., 2020] Ahmadian, S., Joorabloo, N., Jalili, M., Ren, Y., Meghdadi, M., and Afsharchi, M. (2020). A social recommender system based on reliable implicit relationships. *Knowledge-Based Systems*, 192 :105371.
- [Al Jawarneh et al., 2020] Al Jawarneh, I. M., Bellavista, P., Corradi, A., Foschini, L., Montanari, R., Berrocal, J., and Murillo, J. M. (2020). A pre-filtering approach for incorporating contextual information into deep learning based recommender systems. *IEEE Access*, 8 :40485–40498.
- [Allahyari and Kochut, 2016] Allahyari, M. and Kochut, K. (2016). Semantic context-aware recommendation via topic models leveraging linked open data. In *International Conference on Web Information Systems Engineering*, pages 263–277. Springer.
- [Ayub et al., 2019] Ayub, M., Ghazanfar, M. A., Mehmood, Z., Saba, T., Alharbey, R., Munshi, A. M., and Alrige, M. A. (2019). Modeling user rating preference behavior to improve the performance of the collaborative filtering based recommender systems. *PloS one*, 14(8) :e0220129.
- [Azadjalal et al., 2017] Azadjalal, M. M., Moradi, P., Abdollahpouri, A., and Jalili, M. (2017). A trust-aware recommendation method based on pareto dominance and confidence concepts. *Knowledge-Based Systems*, 116 :130–143.
- [Baltrunas et al., 2011] Baltrunas, L., Ludwig, B., and Ricci, F. (2011). Matrix factorization techniques for context aware recommendation. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 301–304.
- [Baltrunas and Ricci, 2009] Baltrunas, L. and Ricci, F. (2009). Context-based splitting of item ratings in collaborative filtering. In *Proceedings of the third ACM conference on Recommender systems*, pages 245–248.
- [Baltrunas and Ricci, 2014] Baltrunas, L. and Ricci, F. (2014). Experimental evaluation of context-dependent collaborative filtering using item splitting. *User Modeling and User-Adapted Interaction*, 24(1-2) :7–34.
- [Bazire and Brézillon, 2005] Bazire, M. and Brézillon, P. (2005). Understanding context before using it. In *International and Interdisciplinary Conference on Modeling and Using Context*, pages 29–40. Springer.
- [Beldjoudi et al., 2016] Beldjoudi, S., Seridi, H., and Benzine, A. (2016). Améliorer la recommandation de ressources dans les folksonomies par l’utilisation de linked open data. In *IC2016 : Ingénierie des Connaissances*.

- [Bell and Koren, 2007] Bell, R. M. and Koren, Y. (2007). Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pages 43–52. IEEE.
- [Berkani et al., 2021] Berkani, L., Belkacem, S., Ouafi, M., and Guessoum, A. (2021). Recommendation of users in social networks : A semantic and social based classification approach. *Expert Systems*, 38(2) :e12634.
- [Berners-Lee et al., 2001] Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. *Scientific american*, 284(5) :34–43.
- [Bizer et al., 2009] Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). Dbpedia-a crystallization point for the web of data. *Journal of web semantics*, 7(3) :154–165.
- [Breese et al., 2013] Breese, J. S., Heckerman, D., and Kadie, C. (2013). Empirical analysis of predictive algorithms for collaborative filtering. *arXiv preprint arXiv :1301.7363*.
- [Brown, 1998] Brown, P. J. (1998). Triggering information by context. *Personal Technologies*, 2(1) :18–27.
- [Burke, 2002] Burke, R. (2002). Hybrid recommender systems : Survey and experiments. *User modeling and user-adapted interaction*, 12(4) :331–370.
- [Byun and Cheverst, 2004] Byun, H. E. and Cheverst, K. (2004). Utilizing context history to provide dynamic adaptations. *Applied Artificial Intelligence*, 18(6) :533–548.
- [Campos et al., 2014] Campos, P. G., Cantador, I., Díez, F., and Fernández-Tobías, I. (2014). A criterion based on fisher’s exact test for item splitting in context-aware recommender systems. In *2014 33rd International Conference of the Chilean Computer Science Society (SCCC)*, pages 80–82. IEEE.
- [Campos et al., 2013] Campos, P. G., Fernández-Tobías, I., Cantador, I., and Díez, F. (2013). Context-aware movie recommendations : an empirical comparison of pre-filtering, post-filtering and contextual modeling approaches. In *International Conference on Electronic Commerce and Web Technologies*, pages 137–149. Springer.
- [Chaari et al., 2005] Chaari, T., Laforest, F., and Flory, A. (2005). Adaptation des applications au contexte en utilisant les services web. In *Proceedings of the 2nd French-speaking conference on Mobility and ubiquity computing*, pages 111–118.

- [Codina and Ceccaroni, 2011] Codina, V. and Ceccaroni, L. (2011). Extending recommendation systems with semantics and context-awareness : Pre-filtering algorithms. In *CCIA*, pages 81–90.
- [Codina et al., 2013] Codina, V., Ricci, F., and Ceccaroni, L. (2013). Exploiting the semantic similarity of contextual situations for pre-filtering recommendation. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 165–177. Springer.
- [Deebak and Al-Turjman, 2020] Deebak, B. and Al-Turjman, F. (2020). A novel community-based trust aware recommender systems for big data cloud service networks. *Sustainable Cities and Society*, 61 :102274.
- [Dey, 2001] Dey, A. K. (2001). Understanding and using context. *Personal and ubiquitous computing*, 5(1) :4–7.
- [Dridi et al., 2020] Dridi, R., Zammali, S., Alsulimani, T., and Arour, K. (2020). Effective rating prediction based on selective contextual information. *Information Sciences*, 510 :218–242.
- [el yebdri et al., 2019] el yebdri, z., Benslimane, S. M., and Lahfa, F. (2019). Trust and context aware splitting approach for improving prediction in recommender system.
- [El Yebdri et al., 2021] El Yebdri, Z., Benslimane, S. M., Lahfa, F., Barhamgi, M., and Benslimane, D. (2021). Context-aware recommender system using trust network. *Computing*, pages 1–19.
- [Fridi and Benslimane, 2017] Fridi, A. and Benslimane, S. M. (2017). Towards semantics-aware recommender system : A lod-based approach. *International Journal of Modern Education and Computer Science*, 9(2) :55.
- [Ghenname, 2015] Ghenname, M. (2015). *Le web social et le web sémantique pour la recommandation de ressources pédagogiques*. PhD thesis, Université Jean Monnet-Saint-Etienne.
- [Gohari et al., 2017] Gohari, F. S., Haghghi, H., and Aliee, F. S. (2017). A semantic-enhanced trust based recommender system using ant colony optimization. *Applied Intelligence*, 46(2) :328–364.
- [Guo et al., 2012] Guo, G., Zhang, J., and Thalmann, D. (2012). A simple but effective method to incorporate trusted neighbors in recommender systems. In *International conference on user modeling, adaptation, and personalization*, pages 114–125. Springer.

- [Guo et al., 2014a] Guo, G., Zhang, J., and Thalmann, D. (2014a). Merging trust in collaborative filtering to alleviate data sparsity and cold start. *Knowledge-Based Systems*, 57 :57–68.
- [Guo et al., 2014b] Guo, G., Zhang, J., Thalmann, D., and Yorke-Smith, N. (2014b). Etaf : An extended trust antecedents framework for trust prediction. In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, pages 540–547. IEEE.
- [Guo et al., 2015] Guo, G., Zhang, J., and Yorke-Smith, N. (2015). Trustsvd : Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.
- [Gupta and Nagpal, 2015] Gupta, S. and Nagpal, S. (2015). Trust aware recommender systems : a survey on implicit trust generation techniques. *International Journal of Computer Science and Information Technologies*, 6(4) :3594–3599.
- [Haydar, 2014] Haydar, C. A. (2014). *Les systèmes de recommandation à base de confiance*. PhD thesis, Université de Lorraine.
- [Herlocker et al., 2002] Herlocker, J., Konstan, J. A., and Riedl, J. (2002). An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information retrieval*, 5(4) :287–310.
- [Herlocker et al., 2017] Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. (2017). An algorithmic framework for performing collaborative filtering. In *ACM SIGIR Forum*, volume 51, pages 227–234. ACM New York, NY, USA.
- [Herlocker et al., 2004] Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1) :5–53.
- [Hull et al., 1997] Hull, R., Neaves, P., and Bedford-Roberts, J. (1997). Towards situated computing. In *Digest of papers. first international symposium on wearable computers*, pages 146–153. IEEE.
- [Jamali and Ester, 2009] Jamali, M. and Ester, M. (2009). Trustwalker : a random walk model for combining trust-based and item-based recommendation. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 397–406.

- [Jamali and Ester, 2010] Jamali, M. and Ester, M. (2010). A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 135–142.
- [Javed et al., 2021] Javed, U., Shaukat, K., Hameed, I. A., Iqbal, F., Alam, T. M., and Luo, S. (2021). A review of content-based and context-based recommendation systems. *International Journal of Emerging Technologies in Learning (iJET)*, 16(3) :274–306.
- [Jiang and Conrath, 1997] Jiang, J. J. and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*.
- [Keikha et al., 2015] Keikha, F., Fathian, M., and Gholamian, M. (2015). Tb-ca : A hybrid method based on trust and context-aware for recommender system in social networks. *Management Science Letters*, 5(5) :471–480.
- [Koren, 2010] Koren, Y. (2010). Factor in the neighbors : Scalable and accurate collaborative filtering. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 4(1) :1–24.
- [Leacock and Chodorow, 1998] Leacock, C. and Chodorow, M. (1998). Combining local context and wordnet similarity for word sense identification. *WordNet : An electronic lexical database*, 49(2) :265–283.
- [Lee and Brusilovsky, 2009] Lee, D. H. and Brusilovsky, P. (2009). Does trust influence information similarity. *Recommender Systems & the Social Web*, 10.
- [Lee et al., 2016] Lee, J.-H. et al. (2016). Improved post-filtering method using context compensation. *International Journal of Fuzzy Logic and Intelligent Systems*, 16(2) :119–124.
- [Lee and Ma, 2016] Lee, W.-P. and Ma, C.-Y. (2016). Enhancing collaborative recommendation performance by combining user preference and trust-distrust propagation in social networks. *Knowledge-Based Systems*, 106 :125–134.
- [Li et al., 2014] Li, J., Sun, C., and Lv, J. (2014). Tcmf : trust-based context-aware matrix factorization for collaborative filtering. In *2014 IEEE 26th international conference on tools with artificial intelligence*, pages 815–821. IEEE.
- [Li et al., 2016] Li, J., Yang, R., and Jiang, L. (2016). Dtcmf : Dynamic trust-based context-aware matrix factorization for collaborative filtering. In *2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*, pages 914–919. IEEE.

- [Li, 2021] Li, S. (2021). *Context-aware recommender system for system of information systems*. PhD thesis, Université de Technologie de Compiègne.
- [Lin et al., 1998] Lin, D. et al. (1998). An information-theoretic definition of similarity. In *Icml*, volume 98, pages 296–304.
- [Liu, 2013] Liu, X. (2013). Towards context-aware social recommendation via trust networks. In *International Conference on Web Information Systems Engineering*, pages 121–134. Springer.
- [Liu, 2014] Liu, X. (2014). *Context-aware recommender systems for implicit data*. PhD thesis, University of York.
- [Louhab et al., 2017] Louhab, F. E., Bahnasse, A., and Talea, M. (2017). Towards a contextual mobile learning deployment : An overview. *International Journal of Computer Science and Network Security*, 17(7) :80–88.
- [Ma et al., 2009] Ma, H., Lyu, M. R., and King, I. (2009). Learning to recommend with trust and distrust relationships. In *Proceedings of the third ACM conference on Recommender systems*, pages 189–196.
- [Maleček et al., 2019] Maleček, L., Balcar, Š., and Peška, L. (2019). Lodbookrec : Linked open data for books recommendation. In *Proceedings of the 9th International Conference on Web Intelligence, Mining and Semantics*, pages 1–6.
- [Marsh and Briggs, 2009] Marsh, S. and Briggs, P. (2009). Examining trust, forgiveness and regret as computational concepts. In *Computing with social trust*, pages 9–43. Springer.
- [Massa, 2007] Massa, P. (2007). A survey of trust use and modeling in real online systems. In *Trust in E-services : Technologies, Practices and Challenges*, pages 51–83. IGI Global.
- [Massa and Avesani, 2004] Massa, P. and Avesani, P. (2004). Trust-aware collaborative filtering for recommender systems. In *OTM Confederated International Conferences " On the Move to Meaningful Internet Systems"*, pages 492–508. Springer.
- [Massa and Avesani, 2007a] Massa, P. and Avesani, P. (2007a). Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 17–24.
- [Massa and Avesani, 2007b] Massa, P. and Avesani, P. (2007b). Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 17–24.

- [Massa and Avesani, 2009] Massa, P. and Avesani, P. (2009). Trust metrics in recommender systems. In *Computing with social trust*, pages 259–285. Springer.
- [Meyer, 2012] Meyer, F. (2012). Recommender systems in industrial contexts. *arXiv preprint arXiv :1203.4487*.
- [Meymandpour and Davis, 2016] Meymandpour, R. and Davis, J. G. (2016). A semantic similarity measure for linked data : An information content-based approach. *Knowledge-Based Systems*, 109 :276–293.
- [Moradi and Ahmadian, 2015] Moradi, P. and Ahmadian, S. (2015). A reliability-based recommendation method to improve trust-aware recommender systems. *Expert Systems with Applications*, 42(21) :7386–7398.
- [Nilashi et al., 2018] Nilashi, M., Ibrahim, O., and Bagherifard, K. (2018). A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques. *Expert Systems with Applications*, 92 :507–520.
- [Ostuni et al., 2013] Ostuni, V. C., Di Noia, T., Di Sciascio, E., and Mirizzi, R. (2013). Top-n recommendations from implicit feedback leveraging linked open data. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 85–92.
- [Otebolaku and Andrade, 2017] Otebolaku, A. M. and Andrade, M. T. (2017). Context-aware personalization using neighborhood-based context similarity. *Wireless Personal Communications*, 94(3) :1595–1618.
- [Panniello et al., 2009] Panniello, U., Tuzhilin, A., Gorgoglione, M., Palmisano, C., and Pedone, A. (2009). Experimental comparison of pre-vs. post-filtering approaches in context-aware recommender systems. In *Proceedings of the third ACM conference on Recommender systems*, pages 265–268.
- [Parvin et al., 2019] Parvin, H., Moradi, P., and Esmaeili, S. (2019). Tcfaco : Trust-aware collaborative filtering method based on ant colony optimization. *Expert Systems with Applications*, 118 :152–168.
- [Peska and Vojtas, 2013] Peska, L. and Vojtas, P. (2013). Enhancing recommender system with linked open data. In *International Conference on Flexible Query Answering Systems*, pages 483–494. Springer.
- [Picot-Clément, 2011] Picot-Clément, R. (2011). *Une architecture générique de Systèmes de recommandation de combinaison d’items : application au domaine du tourisme*. PhD thesis, Université de Bourgogne.

- [Rada et al., 1989] Rada, R., Mili, H., Bicknell, E., and Blettner, M. (1989). Development and application of a metric on semantic nets. *IEEE transactions on systems, man, and cybernetics*, 19(1) :17–30.
- [Resnick et al., 1994] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). Grouplens : An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186.
- [Resnick and Varian, 1997] Resnick, P. and Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, 40(3) :56–58.
- [Resnik, 1995] Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*.
- [Richa and Bedi, 2021] Richa and Bedi, P. (2021). Trust and distrust based cross-domain recommender system. *Applied Artificial Intelligence*, 35(4) :326–351.
- [Ryan et al., 1998] Ryan, N. S., Pascoe, J., and Morse, D. R. (1998). Enhanced reality fieldwork : the context-aware archaeological assistant. In *Computer applications in archaeology*. Tempus Reparatum.
- [Said et al., 2011] Said, A., De Luca, E. W., and Albayrak, S. (2011). Inferring contextual user profiles-improving recommender performance. In *Proceedings of the 3rd RecSys Workshop on Context-Aware Recommender Systems*.
- [Sassi et al., 2017] Sassi, I. B., Mellouli, S., and Yahia, S. B. (2017). Context-aware recommender systems in mobile environment : On the road of future research. *Information Systems*, 72 :27–61.
- [Schafer et al., 2007] Schafer, J. B., Frankowski, D., Herlocker, J., and Sen, S. (2007). Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer.
- [Schilit et al., 1993] Schilit, B. N., Theimer, M. M., and Welch, B. B. (1993). Customizing mobile applications. In *Proceedings USENIX Symposium on Mobile & Location-independent Computing*, volume 9, pages 20–25.
- [Sejwal and Abulaish, 2019] Sejwal, V. K. and Abulaish, M. (2019). Trust and context-based rating prediction using collaborative filtering : a hybrid approach. In *Proceedings of the 9th international conference on web intelligence, mining and semantics*, pages 1–10.

- [Sejwal and Abulaish, 2021] Sejwal, V. K. and Abulaish, M. (2021). Camo : A context-aware movie ontology generated from lod and movie databases. *Multimedia Tools and Applications*, 80(5) :7247–7269.
- [Shambour and Lu, 2012] Shambour, Q. and Lu, J. (2012). A trust-semantic fusion-based recommendation approach for e-business applications. *Decision Support Systems*, 54(1) :768–780.
- [Shani and Gunawardana, 2011] Shani, G. and Gunawardana, A. (2011). Evaluating recommendation systems. In *Recommender systems handbook*, pages 257–297. Springer.
- [Shardanand and Maes, 1995] Shardanand, U. and Maes, P. (1995). Social information filtering : Algorithms for automating “word of mouth”. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 210–217.
- [Sinha et al., 2001] Sinha, R. R., Swearingen, K., et al. (2001). Comparing recommendations made by online systems and friends. *DELOS*, 106.
- [Syvanen et al., 2005] Syvanen, A., Beale, R., Sharples, M., Ahonen, M., and Lonsdale, P. (2005). Supporting pervasive learning environments : adaptability and context awareness in mobile learning. In *IEEE International workshop on wireless and mobile technologies in education (WMTE'05)*, pages 3–pp. IEEE.
- [Taneja and Arora, 2017] Taneja, A. and Arora, A. (2017). Clu-pof-a novel post filtering approach for efficient context aware recommendations. *Procedia computer science*, 122 :834–841.
- [Tang et al., 2013] Tang, J., Hu, X., Gao, H., and Liu, H. (2013). Exploiting local and global social context for recommendation. In *IJCAI*, volume 13, pages 2712–2718. Citeseer.
- [Villegas et al., 2018] Villegas, N. M., Sánchez, C., Díaz-Cely, J., and Tamura, G. (2018). Characterizing context-aware recommender systems : A systematic literature review. *Knowledge-Based Systems*, 140 :173–200.
- [Wang et al., 2019] Wang, M., Wu, Z., Sun, X., Feng, G., and Zhang, B. (2019). Trust-aware collaborative filtering with a denoising autoencoder. *Neural Processing Letters*, 49(2) :835–849.
- [Weiser, 1999] Weiser, M. (1999). The computer for the 21st century. *ACM SIGMOBILE mobile computing and communications review*, 3(3) :3–11.
- [Wu and Palmer, 1994] Wu, Z. and Palmer, M. (1994). Verb semantics and lexical selection. *arXiv preprint cmp-lg/9406033*.

- [Xiao and Benbasat, 2007] Xiao, B. and Benbasat, I. (2007). E-commerce product recommendation agents : Use, characteristics, and impact. *MIS quarterly*, pages 137–209.
- [Xu et al., 2017] Xu, J., Zhong, Y., Zhu, W., and Sun, F. (2017). Trust-based context-aware mobile social network service recommendation. *Wuhan University Journal of Natural Sciences*, 22(2) :149–156.
- [Yadav et al., 2018] Yadav, S., Kumar, V., Sinha, S., and Nagpal, S. (2018). Trust aware recommender system using swarm intelligence. *Journal of computational science*, 28 :180–192.
- [Yang et al., 2015] Yang, S., Peng, Q., and Chen, L. (2015). The bpso based complex splitting of context-aware recommendation. In *International Symposium on Computational Intelligence and Intelligent Systems*, pages 435–444. Springer.
- [Yuan et al., 2010] Yuan, W., Shu, L., Chao, H.-C., Guan, D., Lee, Y.-K., and Lee, S. (2010). Itars : trust-aware recommender system using implicit trust networks. *IET communications*, 4(14) :1709–1721.
- [Zammali et al., 2015] Zammali, S., Arour, K., and Bouzeghoub, A. (2015). A context features selecting and weighting methods for context-aware recommendation. In *2015 IEEE 39th Annual Computer Software and Applications Conference*, volume 2, pages 575–584. IEEE.
- [Zheng et al., 2012] Zheng, Y., Burke, R., and Mobasher, B. (2012). Optimal feature selection for context-aware recommendation using differential relaxation. *Acm Recsys*, 12.
- [Zheng et al., 2013] Zheng, Y., Burke, R., and Mobasher, B. (2013). Recommendation with differential context weighting. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 152–164. Springer.
- [Zheng et al., 2014] Zheng, Y., Burke, R., and Mobasher, B. (2014). Splitting approaches for context-aware recommendation : An empirical study. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pages 274–279.
- [Zheng et al., 2015a] Zheng, Y., Mobasher, B., and Burke, R. (2015a). Carskit : A java-based context-aware recommendation engine. In *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pages 1668–1671. IEEE.
- [Zheng et al., 2015b] Zheng, Y., Mobasher, B., and Burke, R. (2015b). Similarity-based context-aware recommendation. In *International Conference on Web Information Systems Engineering*, pages 431–447. Springer.

[Zheng et al., 2016] Zheng, Y., Mobasher, B., and Burke, R. (2016). Emotions in context-aware recommender systems. In *Emotions and Personality in Personalized Services*, pages 311–326. Springer.

Résumé

Le travail présenté dans ce manuscrit se situe dans le domaine des systèmes de recommandation sensible au contexte (CARS) qui vise à améliorer les systèmes de recommandation traditionnels (SR) en prenant en compte les informations du contexte lors de la prédiction. Cependant, Ces systèmes souffrent de certains défis, tels que le démarrage à froid et la rareté des données. De nouvelles méthodes sont proposées pour surmonter ces problèmes. Nous proposons dans notre travail, trois contributions. La première contribution consiste à pallier les limites de l'approche du découpage sensible au contexte (Context-aware splitting approach (CASA)) qui représente l'une des approches de pré-filtrage les plus efficaces du système de recommandation sensible au contexte. Nous proposons d'ajouter principalement des informations de confiance ainsi sémantique pour améliorer la qualité de prédiction. La deuxième contribution, consiste à proposer une approche hybride intitulée : Approche de post-filtrage sensible au contexte basée sur la confiance, qui utilise des déclarations de confiance comme une information riche, avec la méthode de compensation de contexte qui appartient à l'approche de post-filtrage contextuel. Les résultats des expérimentations révèlent que ces approches améliorent la pertinence des recommandations et surpasse les autres approches non contextuelles de la littérature en terme de précision. Une autre approche que nous proposons : Approche de modélisation contextuelle basée sur la sémantique et la confiance (ST-CAMA), qui combine les informations de confiance et de contexte en utilisant la pondération contextuelle. Aussi, afin de sélectionner uniquement les voisins de confiance et qui ont des intérêts communs à l'item cible pour l'utilisateur actif, nous proposons de construire des clusters se basant sur des similarités sémantiques enrichi sémantiquement via les données ouvertes liées (LOD).

Mots clés : Système de recommandation, contexte, confiance, sémantique, méthode de compensation, système de recommandation sensible au contexte, système de recommandation basé confiance.

Abstract

The work presented in this manuscript is in the area of Context-Aware Recommender Systems (CARS) which aims to improve traditional recommendation systems (SR) by taking into account context information in predicting process. However, SR suffers from some challenges, such as cold start and data sparsity. New methods are proposed to overcome these problems. We propose in our work, three contributions. The first contribution aims to overcome the limitations of the Context-Aware Splitting Approach (CASA), which represents one of the most effective pre-filtering approaches of the context-aware recommender system. We propose to add mainly confidence information as well as semantics to improve the quality of prediction. The second contribution aims to propose a hybrid approach entitled: (Trust based Context aware Post Filtering Approach (TCPoFA)), which belongs to contextual post-filtering approach. The results of the experiments reveal that these approaches improve the relevance of the recommendations and outperform other non-contextual approaches in the literature in terms of precision. Another approach we propose: Contextual modeling approach based on semantics and trust (ST-CAMA), which combines trust and context information using contextual weighting. Also, in order to select only trusted neighbors who have interests common to the target item for the active user, we propose to build clusters based on semantic similarities enriched semantically via linked open data (LOD).

Keywords: Recommender system, context, trust, semantics, compensation method, context-aware recommender system, trust-aware recommender system.

ملخص

ينظم العمل المقدم في هذه الأطروحة إلى مجال أنظمة التوصية الحساسة للسياق (CARS) والتي حسنت أنظمة التوصية التقليدية (SRs) من خلال مراعاة معلومات السياق أثناء التنبؤ. ومع ذلك، فإن هذه الأنظمة تعاني من بعض التحديات، مثل البداية الباردة وندرة البيانات. تم اقتراح طرق جديدة للتغلب على هذه المشاكل. في عملنا، نقتراح ثلاث مساهمات: تتمثل المساهمة الأولى في التغلب على نهج التقسيم الواعي بالسياق (CASA)، والذي يمثل أحد أهم مناهج التوصية الفعال والحساس للسياق. نقتراح بشكل أساسي إلى إضافة معلومات الثقة بالإضافة إلى الدلالات لتحسين جودة التنبؤ. تتمثل المساهمة الثانية في اقتراح نهج مختلط بعنوان: نهج ما بعد التصفية الحساسة للسياق على أساس الثقة، والذي يستخدم بيانات الثقة كمعلومات غنية مع طريقة تعويض السياق التي تنتمي إلى نهج التصفية اللاحق السياقي. تكشف النتائج التجريبية أن هذه المساهمات تحسن نتائج التوصيات وتغلب المناهج التي لا تأخذ بعين الاعتبار السياق من حيث الدقة. مساهمة أخرى نقتراحها: نهج التصميم السياقي القائم على الدلالات والثقة (ST-CAMA)، والذي يجمع بين الثقة ومعلومات السياق باستخدام الترجيح السياقي. أيضًا، من أجل تحديد المستخدمين الموثوق بهم فقط الذين لديهم اهتمامات مشتركة في العنصر المستهدف للمستخدم، نقتراح إنشاء مجموعات استنادًا إلى أوجه التشابه الدلالية التي يتم إثرائها معنويًا عبر البيانات المرتبطة المفتوحة (LOD).

الكلمات المفتاحية: نظام التوصية، السياق، الثقة، الدلالات، طريقة التعويض، نظام التوصية الحساسة للسياق، نظام التوصية الحساسة للثقة.