



HAL
open science

Context-aware recommender system for system of information systems

Siying Li

► **To cite this version:**

Siying Li. Context-aware recommender system for system of information systems. Technology for Human Learning. Université de Technologie de Compiègne, 2021. English. NNT : 2021COMP2602 . tel-03232938

HAL Id: tel-03232938

<https://theses.hal.science/tel-03232938v1>

Submitted on 22 May 2021

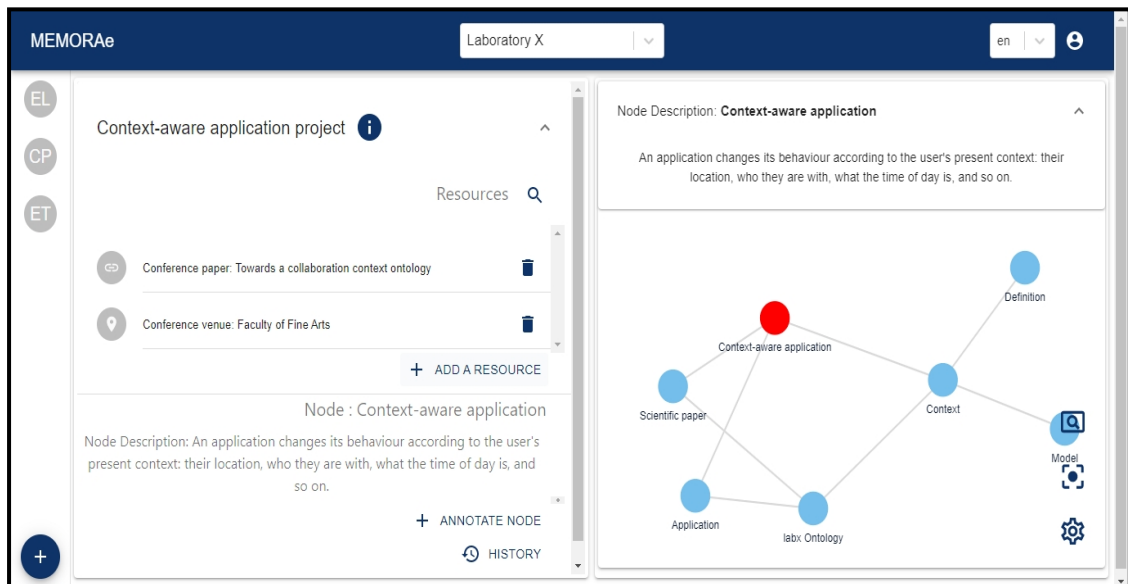
HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Par Siying LI

*Context-aware recommender system for system
of information systems*

Thèse présentée
pour l'obtention du grade
de Docteur de l'UTC



MEMORAe Laboratory X en

Context-aware application project

Resources

- Conference paper: Towards a collaboration context ontology
- Conference venue: Faculty of Fine Arts

+ ADD A RESOURCE

Node : Context-aware application

Node Description: An application changes its behaviour according to the user's present context: their location, who they are with, what the time of day is, and so on.

+ ANNOTATE NODE HISTORY

Node Description: Context-aware application

An application changes its behaviour according to the user's present context: their location, who they are with, what the time of day is, and so on.

Graph visualization showing nodes: Scientific paper, Application, labx Ontology, Context, Definition, Model.

Soutenue le 7 avril 2021

Spécialité : Informatique : Unité de recherche Heudyasic (UMR-7253)

D2602

UNIVERSITÉ DE TECHNOLOGIE DE COMPIÈGNE
CNRS, HEUDIASYC (HEURISTIQUE ET DIAGNOSTIC
DES SYSTÈMES COMPLEXES)



DOCTORAL THESIS

Context-Aware Recommender System for System of Information Systems

Spécialité : Informatique

Author

Siying LI

Supervisors:

Marie-Hélène ABEL

Elsa NEGRE

A thesis submitted in fulfillment of the requirements for the degree of Doctor

07 April, 2021

UNIVERSITÉ DE TECHNOLOGIE DE COMPIÈGNE
CNRS, HEUDIASYC (HEURISTIQUE ET DIAGNOSTIC DES SYSTÈMES
COMPLEXES)

Abstract

Context-Aware Recommender System for System of Information Systems

by Siying LI

Working collaboratively is no longer an issue but a reality, what matters today is how to implement collaboration so that it is as successful as possible. However, successful collaboration is not easy and is conditioned by different factors that can influence it. It is therefore necessary to take these impacting factors into account within the context of collaboration for promoting the effectiveness of collaboration. Among the impacting factors, collaborator is a main one, which is closely associated with the effectiveness and success of collaborations. The selection and/or recommendation of collaborators, taking into account the context of collaboration, can greatly influence the success of collaboration.

Meanwhile, thanks to the development of information technology, many collaborative tools are available, such as e-mail and real-time chat tools. These tools can be integrated into a web-based collaborative work environment. Such environments allow users to collaborate beyond the limit of geographical distances. During collaboration, users can utilize multiple integrated tools, perform various activities, and thus leave traces of activities that can be exploited. This exploitation will be more precise when the context of collaboration is described. It is therefore worth developing web-based collaborative work environments with a model of the collaboration context. Processing the recorded traces can then lead to context-aware collaborator recommendations that can reinforce the collaboration.

To generate collaborator recommendations in web-based Collaborative Working Environments, this thesis focuses on producing context-aware collaborator recommendations by defining, modeling, and processing the collaboration context. To achieve this, we first propose a definition of the collaboration context and choose to build a collaboration context ontology given the advantages of the ontology-based modeling approach. Next, an ontology-based semantic similarity is developed and applied in three different algorithms (i.e., PreF1, PoF1, and PoF2) to generate context-aware collaborator recommendations. Furthermore, we deploy the collaboration context ontology into web-based Collaborative Working Environments by considering an architecture of System of Information Systems from the viewpoint of web-based Collaborative Working Environments. Based on this architecture, a corresponding prototype of web-based Collaborative Working Environment is then constructed. Finally, a dataset of scientific collaborations is employed to test and evaluate the performances of the three context-aware collaborator recommendation algorithms.

Résumé

Travailler en collaboration n'est plus une question mais une réalité, la question qui se pose aujourd'hui concerne la mise en oeuvre de la collaboration de façon à ce qu'elle soit la plus réussie possible. Cependant, une collaboration réussie n'est pas facile et est conditionnée par différents facteurs qui peuvent l'influencer. Il est donc nécessaire de considérer ces facteurs au sein du contexte de collaboration pour favoriser l'efficacité de collaboration. Parmi ces facteurs, le collaborateur est un facteur principal, qui est étroitement associé à l'efficacité et à la réussite des collaborations. Le choix des collaborateurs et/ou la recommandation de ces derniers en tenant compte du contexte de la collaboration peut grandement influencer la réussite de cette dernière.

En même temps, grâce au développement des technologies de l'information, de nombreux outils numériques de collaboration sont mis à la disposition tels que les outils de mail et de chat en temps réel. Ces outils numériques peuvent eux-mêmes être intégrés dans un environnement de travail collaboratif basé sur le web. De tels environnements permettent aux utilisateurs de collaborer au-delà de la limite des distances géographiques. Ces derniers laissent ainsi des traces d'activités qu'ils deviennent possibles d'exploiter. Cette exploitation sera d'autant plus précise que le contexte sera décrit et donc les traces enregistrées riches en description. Il devient donc intéressant de développer les environnements de travail collaboratif basé sur le web en tenant d'une modélisation du contexte de la collaboration. L'exploitation des traces enregistrées pourra alors prendre la forme de recommandation contextuelle de collaborateurs pouvant renforcer la collaboration.

Afin de générer des recommandations de collaborateurs dans des environnements de travail collaboratifs basés sur le web, cette thèse se concentre sur la génération des recommandations contextuelles de collaborateurs en définissant, modélisant et traitant le contexte de collaboration. Pour cela, nous proposons d'abord une définition du contexte de collaboration et choisissons de créer une ontologie du contexte de collaboration compte tenu des avantages de l'approche de modélisation en l'ontologie. Ensuite, une similarité sémantique basée sur l'ontologie est développée et appliquée dans trois algorithmes différents (i.e., PreF1, PoF1 et PoF2) afin de générer des recommandations contextuelles des collaborateurs. Par ailleurs, nous déployons l'ontologie de contexte de collaboration dans des environnements de travail collaboratif basés sur le web en considérant une architecture de système des systèmes d'informations du point de vue des environnements de travail collaboratif basés sur le web. À partir de cette architecture, un prototype correspondant d'environnement de travail collaboratif basé sur le web est alors construit. Enfin, un ensemble de données de collaborations scientifiques est utilisé pour tester et évaluer les performances des trois algorithmes de recommandation contextuelle des collaborateurs.

Acknowledgements

This thesis and all the work related to it were carried out in the laboratory Heudiasyc at the University of Technology of Compiègne (UTC) from October 2017 to present. Huge thanks go to the encouragement and support from my supervisors, colleagues, friends, and family members that help me overcome all the challenges and difficulties.

First and foremost, I owe my greatest debt of gratitude to my beloved country China and its scholarship council (CSC) that backed me up for three years and a half staying in France. At the meantime, I would also like to express my sincerest gratitude to my supervisors: Mrs. Marie-Hélène Abel, professor at the University of Technology of Compiègne, and Ms. Elsa Negre, associate professor at the Paris Dauphine University. They have guided me throughout my thesis with their great patience, dedication, and knowledge respectively in their domains. Their perfect collaborative guidance is a must for the success of this thesis.

Moreover, I want to thank all the members of the laboratory Heudiasyc and colleagues in the CID team, especially Marinela Shehaj, Nathalie Alexandre, Berengere Guermonprez, Dominique Chambelant, Gildas Bayard, Majd Saleh, Gregoryy Wanderley, Xuhong Li, Chenghao Wang, Zheng Tong, and Qing Tang for their help, accompanying, and friendship.

Finally, my deepest appreciation is dedicated to my family, especially my parents, my grand parents, and my boyfriend, who have always supported me. Your love, comprehension and encouragement power my improvement.

Contents

Abstract	ii
Acknowledgements	iv
I Introduction	1
1 Introduction	2
1.1 Background & Problems	2
1.1.1 Social issues	3
1.1.2 Scientific issues	3
1.1.3 Problem statement	4
1.2 Our approaches and contributions	5
1.3 Dissertation organization	6
II State of the Art	8
2 Literature review	9
2.1 Introduction	9
2.2 Collaboration	10
2.2.1 Definition	10
2.2.2 Factors of collaboration	11
2.2.3 Discussion	16
2.3 Context	16
2.3.1 Definition	17
2.3.2 Classification	20

2.3.2.1	User context	22
2.3.3	Modeling	25
2.3.3.1	Key-value model	25
2.3.3.2	Markup model	25
2.3.3.3	Graphical model	26
2.3.3.4	Object-oriented model	26
2.3.3.5	Logic-based model	26
2.3.3.6	Ontology-based model	27
2.3.4	Discussion	28
2.4	Web-based collaborative working environment and system of information systems	28
2.4.1	Web-based collaborative working environment	29
2.4.2	From system of systems to system of information systems	32
2.4.3	Discussion	36
2.5	Recommender systems	36
2.5.1	2D recommender system	37
2.5.1.1	Content-Based filtering	38
2.5.1.2	Collaborative Filtering	39
2.5.1.3	Knowledge-Based	40
2.5.1.4	Hybrid	41
2.5.2	Context-aware recommender system	43
2.5.2.1	Contextual pre-filtering	44
2.5.2.2	Contextual post-filtering	45
2.5.2.3	Contextual modeling	45
2.5.3	Discussion	46
2.6	Chapter summary	47
3	A survey of MEMORAe approach	49
3.1	Introduction	49
3.2	MC2 and E-MEMORAe2.0	49

3.3	SOIS and MEMORAe SoIS	53
3.4	Discussion	56
3.5	Chapter summary	56
III	Contributions	57
4	Collaboration context and its model	58
4.1	Introduction	58
4.2	Definitions	58
4.3	Model: Collaboration context ontology	61
4.4	Chapter summary	69
5	Context-aware collaborator recommendations	70
5.1	Introduction	70
5.2	Preliminary	71
5.2.1	Probabilistic matrix factorization	71
5.2.2	Ontology-based semantic similarity	73
5.3	2D collaborator recommendations	74
5.4	Treatment of the collaboration context	77
5.4.1	Terminology	77
5.4.2	Choosing semantic similarity	79
5.4.3	Computing semantic similarity	80
5.4.3.1	Object	80
5.4.3.2	Predicate	82
5.4.3.3	Calculating semantic similarity in MCC	84
5.5	Context-aware collaborator recommendation algorithms	85
5.6	Chapter summary	89
6	Prototype	91
6.1	Introduction	91

6.2	From an architecture of collaborative System of Information Systems towards an architecture of web-based Collaborative Working Environments . . .	92
6.3	Prototype: MEMORAe CWE	94
6.3.1	Usage	95
6.3.2	Application	103
6.4	Discussion	104
6.5	Chapter summary	105
7	Experiments	107
7.1	Introduction	107
7.2	Dataset	107
7.3	Experiments	112
7.3.1	Evaluation methods	112
7.3.2	Experiment with different percentages of training collaborations . . .	114
7.3.3	Experiment with different values of K	115
7.3.4	Experiment with different sets of scientific collaborations	117
7.3.5	Experiment with different semantic similarities	121
7.3.6	Experiment with different collaborative filtering algorithms	123
7.4	Discussion	125
7.5	Chapter summary	126
IV	Conclusion, Perspectives and Future Work	128
8	Conclusion, perspectives and future Work	129
8.1	Conclusion	129
8.2	Perspectives and Future Work	131
	Publications	134
	Appendix A The detailed gradient descent algorithm	136
	Appendix B The interfaces for accessing and manipulating instances	138

Appendix C The survey of MEMORAe CWE	139
---	------------

Bibliography	143
---------------------	------------

List of Figures

2.1	An example of the user's location represented by Markup model.	25
2.2	An example of the user's location represented by Graphical model.	26
2.3	An example of the user's location represented by Object-oriented model. . . .	26
2.4	An example of the user's location represented by Ontology-based model. . . .	27
2.5	A collaboration framework (Weiseth et al., 2006).	30
2.6	An example of SoIS: Internet.	35
2.7	The incorporation of context in the recommendation process (Adomavicius and Tuzhilin, 2011).	44
3.1	General view of MC2 with its three main modules (Deparis, 2013; Atrash, 2015).	50
3.2	Individual and group module of MC2 (Deparis, 2013).	50
3.3	VCard in MC2 (Atrash, 2015).	51
3.4	Resource module of MC2 (Deparis, 2013).	51
3.5	Activity module of MC2 (Atrash, 2015; Wang, 2016).	52
3.6	The main interface of E-MEMORAe2.0 (Atrash, 2015).	53
3.7	Resource module of SOIS (Saleh, 2018).	54
3.8	Activity module of SOIS (Saleh, 2018).	54
3.9	The main interface of MEMORAe SoIS (Saleh, 2018).	55
3.10	An architecture of collaborative SoISs (Saleh, 2018).	55
4.1	An architecture of context models.	59
4.2	The architecture of the collaboration context ontology.	60
4.3	Eight contextual dimensions of collaboration in MCC (T-Box component). . . .	62
4.4	The dimension <i>Collaborator</i> in MCC (T-Box component).	64

4.5	The goal of the context-aware application project in the scenario (A-Box component).	65
4.6	<i>Time</i> , <i>Relation</i> and <i>Satisfaction</i> representations in the scenario (A-Box component).	65
4.7	Collaborator profile in MCC (T-Box component).	66
4.8	Marinela's profile in the scenario (A-Box component).	66
4.9	The dimension <i>Resource</i> in MCC (T-Box component).	67
4.10	The dimension <i>Activity</i> in MCC (T-Box component).	68
4.11	Emma's creating resources activity (A-Box component).	68
4.12	Relation between dimensions <i>Location</i> and <i>Activity</i> in MCC (T-Box component).	68
5.1	Main phases of PreF and PoF.	70
5.2	Distribution of classes and interrelations according to their role in the semantic 3-uple.	79
5.3	Apply TF-IDF to measure the amount of information provided by predicates.	83
6.1	Considering a web-based CWE as a collaborative SoIS.	92
6.2	The main interface of MEMORAe CWE.	94
6.3	Selecting an organization in MEMORAe CWE.	96
6.4	Organizational knowledge graphs in MEMORAe CWE.	96
6.5	Group profile interface of the user group <i>Context-aware application project</i> .	97
6.6	Marinela's user profile interface in MEMORAe CWE.	98
6.7	Emma's interface in MEMORAe CWE.	99
6.8	Collaborative tools integrated into MEMORAe CWE.	99
6.9	Searching papers through API of HAL.	100
6.10	Detailed information of the conference paper.	100
6.11	Emma's activities in the user group of <i>Context-aware application project</i> .	101
6.12	Emma's activities in the organization <i>Laboratory X</i> .	101
6.13	Emma's external collaborative activities in the organization <i>Laboratory X</i> .	102
6.14	The location of conference venue in MEMORAe CWE.	102

6.15	Partial organizational ontology of the course NF01.	103
7.1	F1 with different percentages of training collaborations in a set of scientific collaborations.	114
7.2	MAE with different percentages of training collaborations in a set of scientific collaborations.	115
7.3	Execution time with different percentages of training collaborations in a set of scientific collaborations.	115
7.4	F1 with different K in a set of scientific collaborations.	116
7.5	MAE with different K in a set of scientific collaborations.	116
7.6	Execution time with different K in a set of scientific collaborations.	117
7.7	F1 with different sets of scientific collaborations.	118
7.8	MAE with different sets of scientific collaborations.	119
7.9	Execution time with different sets of scientific collaborations.	120
7.10	F1 with different semantic similarities in a set of scientific collaborations.	122
7.11	MAE with different semantic similarities in a set of scientific collaborations.	122
7.12	Execution time with different semantic similarities in a set of scientific collaborations.	122
7.13	F1 with different 2D recommendation algorithms in a set of scientific collaborations.	124
7.14	MAE with different 2D recommendation algorithms in a set of scientific collaborations.	124
7.15	Execution time with different 2D recommendation algorithms in a set of scientific collaborations.	124

List of Tables

2.1	Three types of factors (San Martín-Rodríguez et al., 2005).	13
2.2	Seven groups of factors (Patel, Pettitt, and Wilson, 2012).	13
2.3	Six categories of factors (Mattessich and Monsey, 1992).	14
2.4	Summary of factors related to collaborator.	15
2.5	Summary of contextual factors.	19
2.6	Classification of human-factors-related context (Schmidt, Beigl, and Gellersen, 1999).	19
2.7	Classification of entities (Zimmermann, Lorenz, and Oppermann, 2007).	20
2.8	Five contextual dimensions of an entity (Zimmermann, Lorenz, and Oppermann, 2007).	22
2.9	Summary of 2D recommendation approaches that can be directly used in CARS methods.	46
7.1	Fields of academic articles in the dataset.	108
7.2	Article numbers in each block.	109
7.3	Transformation from articles' side information into the contexts of scientific collaborations.	110
7.4	Classification of collaborators in generate recommendations for an author u in a testing collaboration c .	113
7.5	Summary of experiments.	127
B.1	Interface for the instances.	138
B.2	Interface for the lists of instances.	138

List of Abbreviations

2D RS	2-Dimensional Recommender System
AdaMF	context- Adaptive Matrix Factorization
BPMF	Bayesian Probabilistic Matrix Factorization
CAMF	Context-Aware Matrix Factorization
CARS	Context-Aware Recommender System
CB	Content- Based filtering
CF	Collaborative Filtering
CM	Contextual Modeling
CMF	Collective Matrix Factorization
CML	Context Modeling Language
CoOL	Context Ontology Language
CSCW	Computer Supported Collaborative Work
CWE	Collaborative Working Environment
ICT	Information and Communication Technology
KB	Knowledge-Based
LDA	Latent Dirichlet Allocation
MAE	Mean Absolute Error
MAG	Microsoft Academic Graph
MF	Matrix Factorization
NCF	Neural network-based Collaborative Filtering
PMF	Probabilistic Matrix Factorization
PoF	contextual Post-Filtering
PreF	contextual Pre-Filtering
RecSys	ACM Recommender Systems conference
RS	Recommender System
SLIM	Sparse LInear Model
SLMC	Seven Layer Model of Collaboration
SoIS	System of Information Systems
SoS	System of Systems
SVD	Singular Value Decomposition
ROA	Resource-Oriented Architecture
TF	Tensor Factorization
TF-IDF	Term Frequency-Inverse Document Frequency

UML	Unified Modeling Language
VBE	Virtual organization Breeding Environment
VO	Virtual Organization
VSM	Vector Space Model

Dedicated to my parents, grand parents, and loved ones.

Part I

Introduction

Chapter 1

Introduction

1.1 Background & Problems

Collaboration is efficient way of working, where a group of people (at least two) work together to achieve common goals (e.g., creating or producing something) (Mattessich and Monsey, 1992). It has received much attention and has had a significant impact. So far, collaboration has benefited people in many fields (e.g., sports, business, academia, technology, education, arts, and even politics), especially when the collaboration is successful. However, successful collaboration is often difficult to achieve (Patel, Pettitt, and Wilson, 2012). There are various factors that can influence collaboration and its success. Among these factors, *collaborator* can significantly influence the effectiveness and success of collaborations. Appropriate collaborators can contribute to solving the problems encountered in collaborations and creating new collaborations (Doherty, 2015). This then increases the efficiency of collaboration and allows a positive relationship to be established between its members. Hence, we are interested in recommending such collaborators to people for collaborations.

However, recommended collaborators are not always relevant to people's needs, especially when they are collaborating in web-based Collaborative Work Environments (CWEs). Integrating different collaborative tools (e.g., instant messaging and resource management tools), a web-based CWE is a computer-supported software enabling people to collaborate without any geographical distance limits (Laso Ballesteros and Prinz, 2008; Martínez-Carreras et al., 2007; Li, Abel, and Barthès, 2012). In such environments, there is a lot of information that can influence people's needs of collaborators but is not considered in generating collaborator recommendations, such as information about people's activities during collaborations. Besides, this information comes from heterogeneous tools, increasing challenges to process and apply it for generating collaborator recommendations.

Therefore, one of the current issues is how to generate relevant collaborator recommendations in web-based CWEs, which is the main concern of this thesis. Specifically, we introduce this issue from two aspects: social and scientific. Then, the problem statement is presented.

1.1.1 Social issues

During collaborations, people are often expected to be geographically close. This, in turn, creates a geographical distance limit for collaborations. To overcome such a limitation, many technologies have been developed, such as computer technology (Neale, Carroll, and Rosson, 2004). This leads to the emergence of Computer Supported Collaborative Work (CSCW) field that combines the understanding of the way people work in groups with the enabling technologies of computer networking (Wilson, 1991). In this field, various collaborative supports (e.g. hardware, software, services, and techniques) are developed rapidly, which offer people with convenience in terms of tools and environments during their collaborations (Mattessich and Monsey, 1992; Patel, Pettitt, and Wilson, 2012). Particularly, these tools can be integrated into a web-based CWE to support people's collaborations.

But simultaneously, the increasing appearance of collaborative tools also raise difficulties for people's collaborations in web-based CWEs. For example, the heterogeneous and disorganized information in distinct tools can weaken people's effectiveness, prevent them from achieving the desired goals of collaboration, thus make collaborations difficult to succeed (Kotlarsky and Oshri, 2005). Thus, while generating collaborator recommendations in web-based CWEs, we need to consider various factors that can influence the success of collaborations, such as *goals* of collaboration, utilized *resources*, and *collaborators* themselves (Patel, Pettitt, and Wilson, 2012; Mattessich and Monsey, 1992; Oliveira, Tinoca, and Pereira, 2011).

However, it is insufficient to just deal with the impacting factors of collaboration. The context can also influence people's actions, effectiveness, and thus the success of collaborations. Here, the context includes various characteristic information of a collaboration, such as the outcomes generated within collaborations. Therefore, in addition to concentrating only on these factors, considering them within the context of collaborations is necessary to form and succeed collaborations (Taylor-Powell and Rossing, 2009). In other words, the collaboration context should be taken into account to generate relevant collaborator recommendations in web-based CWEs.

1.1.2 Scientific issues

In web-based CWEs, users (i.e., people) are able to collaborate in groups, regardless of their geographical distance (Carreras and Skarmeta, 2006; Su et al., 2005; Kan, Duffy, and Su, 2001; Bafoutsou and Mentzas, 2002). While collaborating, they can apply different collaborative tools that are integrated and deployed in web-based CWEs, allowing them to obtain services and/or use resources. Particularly, each tool is an autonomous and independent information system. Together with the web-based CWE, they form a network of autonomous and independent information systems that are integrated together to accomplish

common missions, known as a System of Information Systems (SoIS) (Karcianas and Hesami, 2011; Saleh and Abel, 2016). Specifically, a web-based CWE with its integrated tools can be considered as a collaborative SoIS, where these independent tools work together to support users' collaborations.

In such environments, the aggregation of different tools results in a significant increase in the amount of information, causing information overload problems for users. Besides, the information from different tools is heterogeneous. This leads to other problems in web-based CWEs, like poor interoperability between tools and information processing difficulties (Carreras and Skarmeta, 2006; Martínez-Carreras et al., 2007). It also creates obstacles for users to organize information about collaborations. This implies that users struggle in retrieving information about collaborators, and need collaborator recommendations in the web-based CWEs.

Hence, recommender systems (RS) are designed and applied, which provide suggestions for users among massive information (Ricci, Rokach, and Shapira, 2015). For example, when a user wants a collaborator for web application development, he/she may receive recommendations of collaborators that have built applications. Such recommendations are intended to aid users in seeking out and choosing collaborators. However, they are sometimes not consistent with users' needs. To produce more relevant recommendations, the context is incorporated into recommendation generation processes, which gave rise to a new branch of RS: context-aware recommender system (CARS) (Adomavicius and Tuzhilin, 2011). Particularly, based on Dey (2001), the context in collaborations indicates any characteristic information that can be used to describe the situation of an entity, which can be a user, object, or event related to a collaboration (including the collaboration itself).

Thus, to generate relevant collaborator recommendations in web-based CWEs, this thesis focuses on building a CARS to produce context-aware collaborator recommendations. To do so, the heterogeneous information in web-based CWEs should be organized and processed within the context of users' collaborations. However, so far, context is still lacking a universal definition, resulting in its diverse models (Bazire and Brézillon, 2005). Hence in web-based CWEs, an appropriate definition and model of the collaboration context are needed, so that the collaboration context can be used to generate context-aware collaborator recommendations.

1.1.3 Problem statement

As mentioned in the background of this thesis, in collaborations, appropriate collaborators are needed, which also affect the effectiveness and the success of collaborations. Thus, it is worth recommending such collaborators to people during collaborations, especially when these people are collaborating in web-based CWEs. This requires us to construct a CARS that generates context-aware collaborator recommendations for them in web-based CWEs.

To this end, the following problems have to be addressed:

- Q1. What is collaboration context and how to model it?
- Q2. How to process and apply the collaboration context in algorithms for generating context-aware collaborator recommendations?
- Q3. How to implement and integrate the collaboration context model into web-based CWEs?
- Q4. Are our context-aware collaborator recommendations algorithms relevant in terms of accuracy and time efficiency?

Therefore, this thesis concerns on defining, modeling, and utilizing the collaboration context in web-based CWEs. Besides, we attempt to build a CARS for producing context-aware collaborator recommendations.

1.2 Our approaches and contributions

To deal with the problems listed in Section 1.1.3, the following pieces of work took place:

- **Define collaboration context and construct a collaboration context ontology.**

We first develop a definition of collaboration context by analyzing and supplementing the previous definitions of context and collaboration. Then, considering the advantages of the ontology-based modeling approach in terms of flexibility, interpretability, and supporting interoperability between different information systems, we decide to construct an ontology-based collaboration context model (i.e. a collaboration context ontology) in web-based CWEs. This model reuses and extends two existing ontologies of the MEMORAe approach (Atrash, Abel, and Moulin, 2014; Wang, 2016; Saleh and Abel, 2016).

- **Develop and employ a semantic similarity in context-aware recommendation algorithms.**

To process the collaboration context, we propose an ontology-based semantic similarity, permitting us to measure the likeness/relatedness between two collaborations in the collaboration context ontology. Then, to generate context-aware collaborator recommendations, this semantic similarity is applied in recommendation algorithms, which are implemented following two methods: contextual pre-filtering (PreF) and post-filtering (PoF) (Adomavicius and Tuzhilin, 2011; Adomavicius et al., 2005; Panniello and Gorgoglione, 2012).

- **Develop the collaboration context ontology in an architecture of web-based CWEs and develop a corresponding prototype.**

Having built the collaboration context ontology, it is necessary to explore how to deploy and implement it in a web-based CWE. For this, we make use of the relationship between web-based CWEs and SoISs. The architecture of a collaborative SoIS (Saleh and Abel, 2016) is therefore considered from the viewpoint of web-based CWEs, so that the collaboration context ontology can organize heterogeneous information in a web-based CWE. Based on this architecture, we also build a corresponding prototype of web-based CWE, which is a continuation of the MEMORAe approach.

- **Test and evaluate the performances of the context-aware recommendation algorithms.**

With the collaboration context ontology and the context-aware recommendation algorithms, the collaboration context can be processed to generate context-aware collaborator recommendations. To test the performances of these algorithms, experiments are carried out on a public dataset of scientific collaborations, which is extracted from multiple academic bibliographies. The results obtained from this dataset are compared and evaluated from two aspects: accuracy and time efficiency.

Besides, to illustrate the usage of the collaboration context ontology and our web-based CWE prototype, the following scenario is utilized in this thesis:

Emma is a PhD student on computer science in Laboratory X. Her thesis is about context-aware applications and guided by two supervisors, Elsa and Marie. During her thesis, Emma has collaborated with several people in different scientific projects. Particularly, Emma collaborated with two engineers, Lucie and Marinela, to build a context-aware application (01/11/2020 - 31/12/2020) and to publish their results in a scientific paper. This is a part of her PhD research.

Lucie is a young engineer with a master's degree in computer science. She finished an internship on constructing interfaces of a web application for a company. Then she started her work in Laboratory X with the context-aware application project.

Marinela is an experienced senior engineer in Laboratory X. She has worked for 5 years and helped a lot of students and researchers on their projects. She is an expert in a variety of programming languages, especially in web applications.

1.3 Dissertation organization

The rest of this thesis is organized as follows:

Part II State of the Art is dedicated to analyzing and comparing previous studies related to our problem. It consists of two chapters:

Chapter 2 Literature review: We begin by providing some background on collaboration and explore various factors that can have an impact on collaborations. We also present the notion *context* from its definition, classification, and modeling approach. Then we focus on the relationship between web-based CWEs and SoISs. The final part is mainly about recommender system and context-aware recommender system.

Chapter 3 A survey of MEMORAe approach: This chapter introduces the MEMORAe approach, made up of a web platform and a core ontology.

Part III Contributions is composed of four chapters.

Chapter 4 Collaboration context and its model: Firstly, we present a definition of collaboration context. Then, based on two existing ontologies of the MEMORAe approach, an ontology-based collaboration context model is constructed, also known as a collaboration context ontology.

Chapter 5 Context-aware collaborator recommendations: This chapter starts by explaining how to calculate an ontology-based semantic similarity to process the information in the collaboration context ontology. Then, the semantic similarity is applied in context-aware collaborator recommendation algorithms, based on two methods: PreF and PoF.

Chapter 6 Prototype: We first deploy the collaboration context ontology into web-based CWEs by considering an architecture of collaborative SoISs from the viewpoint of web-based CWEs. Then a corresponding prototype of web-based CWE is demonstrated.

Chapter 7 Experiments: In this chapter, we utilize a dataset to test and evaluate the performances of our context-aware recommendation algorithms (cf. Chapter 5). Based on the results, we discuss advantages and disadvantages of each recommendation algorithm.

Part IV Conclusion, Perspectives and Future Work comprises only one chapter. Chapter 8 contains the conclusions of our work and the perspectives for future work.

Publications: Finally, the publications related to this thesis are presented.

Part II

State of the Art

Chapter 2

Literature review

2.1 Introduction

As an effective way of working together, collaboration is important in human society (Mattessich and Monsey, 1992). At the same time, collaboration also raises challenges to its participants. They may have problems during collaboration process, such as communication barriers between people speaking different languages. When these problems are inappropriately addressed, both the future process and outcomes of a collaboration suffer, which in turn can affect the efficiency of the collaboration and make it unsuccessful. In other words, collaboration is neither easy to achieve nor guaranteed to succeed, since there are many factors which can influence it (Patel, Pettitt, and Wilson, 2012). Thus, we explore collaboration in terms of its definitions and impacting factors.

In this chapter, we introduce the notion, web-based Collaborative Working Environment (CWE), which has emerged to identify the key elements for collaborators in their daily work (Carreras and Skarmeta, 2006). Another type of system, System of Information Systems (SoIS), is also presented and explained, which can provide inspirations on organizing heterogeneous information in web-based CWEs.

Finally, we focus on recommender systems (RSs). Typical RSs seek to predict how a user would value the utility of an item. By regarding collaborators as items, RSs can also be adapted to generate collaborator recommendations for users in web-based CWEs.

Thus, the remaining parts of this chapter are organized as follows: Section 2.2 presents the notion collaboration and identify what factors can influence its success. Then the definition, classification and modeling of context are introduced in Section 2.3. Section 2.4 mainly covers web-based CWE, SoIS and their relationships. We also illustrate the state of the art of RS and CARS in Section 2.5, including their core functionalities, approaches and techniques in use. Finally, a conclusion is provided at the end of this chapter.

2.2 Collaboration

2.2.1 Definition

As a widely used notion, many definitions of collaboration are presented in the literature. Collaboration indicates the action of working with another person or group of people to create or produce something¹. This word also means the situation of two or more people working together to create or achieve the same thing². While in Merriam-Webster Dictionary online, the noun **collaboration** comes from intransitive verb **collaborate** that is defined as working jointly with others or together especially in an intellectual endeavor³.

From these definitions in dictionaries, one common and fundamental meaning of collaboration is to work together with others (at least one person), which is also its initial significance (Wang, 2016). Moreover, the first two definitions specify the objectives of collaboration: to create, achieve or produce something, while Merriam-Webster Dictionary emphasizes the intellectual attempt to achieve a goal. Therefore, another meaning of collaboration is to accomplish its goals by attempts.

Besides, Patel, Pettitt, and Wilson (2012) summarized that *collaboration involves two or more people engaged in interaction with each other, within a single episode or series of episodes, working together towards the common goals*. Suto and Patitad (2015) highlighted communication process and the functionalities of knowledge transfer in collaborations.

These two definitions focus on interactions (e.g., communication) between collaborators that aim to advance towards goals of collaboration. Here, collaborator can be individuals, groups of people or even organizations. Their actions are carried out by human actors contributing to the collaboration as individual members, or members belonging to the group collaborators or organization collaborators. Moreover, Patel, Pettitt, and Wilson (2012) underlined the goals of collaboration are common and shared by all collaborators.

Therefore, after analyzing all above definitions, a collaboration happens between two or more collaborators and comprises a set of human actors' actions on behalf of the corresponding collaborator to achieve their shared goals. Through collaborations, they can improve their relationships and skills (Doherty, 2015) by interacting with others. Besides, collaboration can also help to (Doherty, 2015):

- a. Find solutions that allow for mutual benefit among collaborators

¹It is based on the definition of collaboration in Oxford Dictionaries online (2019) <https://www.lexico.com/en/definition/collaboration> and Oxford Learner's Dictionaries online (2019) <https://www.oxfordlearnersdictionaries.com/definition/english/collaboration>.

²Cambridge Dictionary online, 2019, <https://dictionary.cambridge.org/dictionary/english/collaboration>

³Merriam-Webster Dictionary online, 2019, <https://www.merriam-webster.com/dictionary/collaboration>

- b. Improve decision-making and problem-solving
- c. Increase ability to adapt to changing environment
- d. Increase ability to resolve/prevent conflicts among collaborators
- e. Increase desire to collaborate among collaborators
- f. Increase interdependence among collaborators
- g. Increase reliance among collaborators
- h. Expand resource base (information, problem-solving options, knowledge, etc.)
- i. Broaden information/perspective on system issues
- j. Increase opportunities for efficiency
- k. Decrease individual risk
 - l. Increase risk sharing among collaborators
- m. Increase ability to handle uncertainty

Nevertheless, these benefits cannot be obtained through any collaborations. Some are inherent to successful collaborations, such as e and f (Doherty, 2015). Based on Kotlarsky and Oshri (2005), the only condition of a successful collaboration is to achieve its desired goals through effort, which can be qualified by product success and personal satisfaction. But evaluating a successful collaboration involves much more than that. Various factors that can influence collaborations must also be considered.

2.2.2 Factors of collaboration

Numerous studies have been implemented to identify different factors that have an impact on collaboration, regardless of whether it succeeds or not. Briggs et al. (2009) proposed a Seven-Layer Model of Collaboration (SLMC) to assist designers of collaborative systems. SLMC defines seven groups of impacting factors in collaborations, one group per layer (Briggs et al., 2009):

- **Goal:** A goal is a desired state or outcome, including factors like motivation, commitment, and other goal-related factors.
- **Product:** A product is a tangible or intangible artifact or outcome produced by collaborators' effort, containing quality, effectiveness and efficiency of products.
- **Activity:** Activities are subtasks that, when completed, produce the outputs that make up the achievement of the group's objective, related to sequences of steps to constitute decision-making and problem-solving approaches.

- **Pattern of Collaboration:** Collaborative patterns are observable patterns of behavior and outcomes that emerge over time in teamwork, depending on regularities of collaborators' actions and outcomes.
- **Technique:** A collaboration technique is a reusable procedure for invoking useful interactions between collaborators working towards a shared goal, for example, brainstorming.
- **Tool:** Collaboration tools are artifacts or devices used in the execution of an operation to advance towards its goals, including designing, developing, deploying and using technologies to support collaboration.
- **Script:** A script is everything team members tell each other and do with their tools to get closer to the collaboration's goals. It can be an internal or external, physical or electronic document.

The seven layers are related to each other. Collaborators apply tools and techniques to carry out activities (set of actions) to create products or achieve shared goals of collaboration. In this process, various scripts are generated and collaboration patterns are established. Together, these form the core of collaboration and allow it to be characterized (Briggs et al., 2009).

Particularly for successful collaborations, Wouters et al. (2017) specified four prerequisites: 1) A shared objective between the stakeholders involved; 2) A synchronization of actions; 3) An exchange of information, between the right entities, at the right time; 4) Complementarity between skills.

Besides, some researches only focus on the factors of successful collaborations.

- Hara et al. (2003) concluded four factors for scientific collaborations: 1) Personal compatibility; 2) Research work connections; 3) Incentives; 4) Socio-technical infrastructure.
- Bruneel, d'Este, and Salter (2010) investigated University-Industry collaborations and concentrated on three factors: 1) Experience of collaboration; 2) Breadth of interaction channels; 3) Inter-organizational trust.
- Camarinha-Matos and Afsarmanesh (2006) summarized principal elements in Virtual Organization Breeding Environment (VBE⁴) that is a subtype of collaborative network⁵ (Afsarmanesh et al., 2009). The elements are: 1) VBE; 2) VBE member; 3)

⁴VBE is a breeding environment with the main goal of increasing both VBE members' chances and preparedness of collaboration in potential Virtual Organizations (VOs) (Camarinha-Matos and Afsarmanesh, 2008). Here, VO is an alliance comprising a set of (legally) independent organizations that share their resources and skills, to achieve their common mission/goal.

⁵A collaborative network is constituted by a variety of entities (e.g., organizations and people) that are largely autonomous, geographically distributed, and heterogeneous in terms of their: operating environment, culture, social capital, and goals (Camarinha-Matos and Afsarmanesh, 2005). Interactions among these entities are supported by computer network, which allows them to collaborate.

VO; 4) Profile; 5) History; 6) Evidence; 7) Bag of assets; 8) Management system; 9) VBE governance; 11) Value system; 12) Trust system.

Occasionally, many factors of successful collaboration are too specific that they have to be clustered into groups/categories. For example, San Martín-Rodríguez et al. (2005) considered three types of factors for interprofessional collaboration in health care (see Table 2.1).

TABLE 2.1: Three types of factors (San Martín-Rodríguez et al., 2005).

Type	Factors
Interactional factors	Willingness to collaborate; Trust; Communication; Mutual respect.
Organizational factors	Organizational structure; Organization's philosophy; Administrative support; Team sources; Coordination and communication mechanisms.
Systemic factors	The social system; The professional system; The educational system.

Patel, Pettitt, and Wilson (2012) applied seven groups of factors to frame a collaborative working model (see Table 2.2).

TABLE 2.2: Seven groups of factors (Patel, Pettitt, and Wilson, 2012).

Group	Factors
Context	Culture; Environment; Business climate; Organizational structure.
Support	Tools; Networks; Training; Team building; Knowledge management; Error management
Tasks	Type; Structure; Demands.
Interaction Processes	Learning; Coordination; Communication; Decision-making.
Team	Roles; Relationships; Shared awareness/knowledge; Common ground; Group processes; Composition.
Individuals	Skills; Psychological factors; Wellbeing.
Overarching Factors	Trust; Conflict; Experience; Goals; Incentives; Time; Constraints; Management; Performance.

Besides, Mattessich and Monsey (1992) figured out 19 factors of successful collaboration and sort them into six categories (see Table 2.3).

TABLE 2.3: Six categories of factors (Mattessich and Monsey, 1992).

Category	Factors
Environment	History of collaboration in the community; Collaborative group seen as a leader in the community; Political/social climate favorable.
Membership	Mutual respect, understanding and trust; Appropriate cross-section of members; Members: collaboration as in theirself-interest; Ability to compromise.
Process/Structure	Members share a stake in both process and outcome; Multiple layers of decision-making; Flexibility; Development of clear roles and policy guidelines; Adaptability.
Communication	Open and frequent communication; Established informal and formal communication links.
Purpose	Concrete, attainable goals and objectives; Shared vision; Unique purpose.
Resources	Sufficient funds; Skilled convener.

All these works mentioned above seek to enhance the success of collaborations. Each concerns on the impacting factors of collaboration in its own field of research, which is why these factors are so diverse. Sometimes, certain factors are not universal and cannot qualify collaborations in other fields. For example, *research work connections* (Hara et al., 2003) is only suitable for scientific collaborations. In commercial collaborations, this factor is no longer adoptable. Based on it, a corresponding factor should be *Business connections* to describe the matching and complementarity of collaborators. Other similar factors include *inter-organizational trust* (Bruneel, d'Este, and Salter, 2010) and *VBE* (Camarinha-Matos and Afsarmanesh, 2006).

Moreover, all research methods in these works can be divided into two main classes: 1) directly explore impacting factors, e.g., (Hara et al., 2003), (Bruneel, d'Este, and Salter, 2010) and (Camarinha-Matos and Afsarmanesh, 2006); 2) integrate factors into different levels by grouping/categorizing, e.g., (Briggs et al., 2009), (San Martín-Rodríguez et al., 2005), and (Patel, Pettitt, and Wilson, 2012). Such a difference indicates that each factor has a specific impact on collaboration. And its impact depends on the domain of collaboration.

Besides, many shared factors that imply the common characteristic of collaboration are summarized and shown below:

- **Goal:** This is essential for collaborations and is mentioned by the definitions of collaboration in Section 2.2.1. A goal can refer to desired outcomes, products, or states but must be shared by collaborators (Briggs et al., 2009; Wouters et al., 2017; Patel, Pettitt, and Wilson, 2012; Mattessich and Monsey, 1992). If the desired goal is attained, this

collaboration is successful (Kotlarsky and Oshri, 2005).

- **Collaborators:** This is also basic for collaboration, which signifies members of collaboration. In some cases, it can be represented by a group of factors that are permanent to collaborators as Patel, Pettitt, and Wilson (2012) and Mattessich and Monsey (1992) do. These related factors are shown in Table 2.4.

TABLE 2.4: Summary of factors related to collaborator.

Factors	References
Relationships	Camarinha-Matos and Afsarmanesh (2006) Patel, Pettitt, and Wilson (2012) Mattessich and Monsey (1992)
Abilities/Skills	Camarinha-Matos and Afsarmanesh (2006) Patel, Pettitt, and Wilson (2012) Mattessich and Monsey (1992)
Sharing and compatibility	Wouters et al. (2017) Hara et al. (2003) Patel, Pettitt, and Wilson (2012) Mattessich and Monsey (1992)
Experiences/Histories	Bruneel, d’Este, and Salter (2010) Camarinha-Matos and Afsarmanesh (2006) Patel, Pettitt, and Wilson (2012) Mattessich and Monsey (1992)
Trust	Bruneel, d’Este, and Salter (2010) Camarinha-Matos and Afsarmanesh (2006) San Martín-Rodríguez et al. (2005) Patel, Pettitt, and Wilson (2012) Mattessich and Monsey (1992)

- **Collaborators’ actions:** This indicates the actions that collaborators undertake to progress towards the goals of collaboration. This consists of interactions/communications among multiple collaborators, or coordination for sub-tasks in collaborations, or decision-making (Briggs et al., 2009; Wouters et al., 2017; Bruneel, d’Este, and Salter, 2010; San Martín-Rodríguez et al., 2005; Patel, Pettitt, and Wilson, 2012; Mattessich and Monsey, 1992).
- **Resources:** This covers various forms of assets used to support collaborations, such as funding, documents, tools, technologies, materials (Briggs et al., 2009; Wouters et al., 2017; San Martín-Rodríguez et al., 2005; Patel, Pettitt, and Wilson, 2012; Mattessich and Monsey, 1992).

- **Context:** This involves the surroundings of collaboration that do not define the collaboration but are closely tied to and influence it. It can be a cultural, organizational (San Martín-Rodríguez et al., 2005) or environmental factor (Mattessich and Monsey, 1992; Patel, Pettitt, and Wilson, 2012). For example, if students learn collaboratively on a site, then the site belongs to this kind of factor when serving as an environment in collaborations. And the site can have an effect on students' actions, patterns, and effectiveness of their learning. Specifically, the site is also termed as a Collaborative Working Environment (CWE), which can support collaborations among collaborators geographically dispersed.

While these factors have some similarity and differences, they complement each other to some extent. Jointly, they describe the features of collaboration and its success. Therefore, in turn, such factors can contribute to assessing the state of collaboration and suggesting adjustments accordingly.

2.2.3 Discussion

To conclude, a collaboration must have three necessary components: at least two collaborators, a common goal, and interactions among collaborators to advance it. During its process, collaborators can derive some benefits, especially when the collaboration is successful. However, successfully collaborating is often uneasy. Various factors can influence collaboration and its success, each having different impacts. This indicates that the success of a collaboration is not determined by a single factor, but by the integration of all these factors. In turn, adjusting these factors can directly or indirectly assist collaboration to succeed. However, none of the previous research has shown any interest in this.

Moreover, goals, collaborators, their actions, resources, and context are the key factors of collaboration, which implies common features of collaboration from the above analysis. Particularly, the factor context may also influence other factors of collaboration, such as collaborators' interactions and their used tools. This complicates how to integrate these factors together.

Besides, since a collaborator plays an essential role in collaborations, recommending appropriate collaborators to people will also help them to collaborate successfully.

2.3 Context

As a complex notion (Adomavicius and Jannach, 2014), context is the keystone that enables any intelligent entity to understand how occurrences in the surrounding world influence its own behavior. This is because context supports intelligent entities to deduce possible

activities and information needs, allowing them to apply appropriate behavior accordingly (Kofod-Petersen and Cassens, 2005).

In the field of information technology, it is essential for human-computer interactions by presenting surrounding facts that make sense (Schmidt, Beigl, and Gellersen, 1999). Context also plays a significant role in establishing a common understanding of service interoperability in distributed systems (Strang and Linnhoff-Popien, 2003).

Besides, context is applied in context-aware applications to provide relevant information and/or services to the user, where relevancy depends on the user's task (Dey, 2001) (e.g., context-aware recommender system). The functions of such applications are assigned to three categories (Dey, Abowd, and Salber, 2001): 1) the presentation of information⁶; 2) the execution of services⁷; 3) the storage of context information attached to other captured information for later retrieval⁸. In these applications, one core accessor is a well-designed context model (Strang, Linnhoff-Popien, and Frank, 2003).

However, context stays at an ill-defined stage (Bazire and Brézillon, 2005). Its various definitions result in diverse contextual information implemented in different context-aware applications. This, in turn, poses difficulties when processing contextual information: it is hard to describe context and relationships in a precise and traceable manner (Strang, Linnhoff-Popien, and Frank, 2003).

Therefore, we first explore definitions of context. Given the diversity of contextual information, we also discuss its classification to categorize it and to comprehend in a systematic manner (Dey, Abowd, and Salber, 2001). Finally, approaches to model context are introduced and summarized.

2.3.1 Definition

In Oxford Dictionaries online (2019)⁹, the notion **context** is defined as the circumstances that form the setting for an event, statement, or idea, and in terms of which it can be fully understood. Psychologically, context is the set of circumstances that frames an event or an object (Bazire and Brézillon, 2005).

Both definitions describe context using the term **circumstance**, which indicates a fact or condition connected with or relevant to an event or action¹⁰. Besides, context is restricted to shaping an event, action or object to enhance others' comprehension for it (the event, action

⁶*Presenting information and services* refers to applications that either present context information to the user or use context to propose appropriate selections of actions to the user.

⁷*Automatically executing a service* describes applications that trigger a command or reconfigure the system on behalf of the user according to context changes.

⁸To *attach context information for later retrieval*, context-aware applications tag captured data with relevant context information.

⁹<https://www.lexico.com/en/definition/context>

¹⁰Oxford Dictionaries online (2019) <https://www.lexico.com/en/definition/circumstance>

or object). Thus, context can be considered as a set of conditions that are pertinent to or constitute an event, statement, or object.

Apart from listed above, context has received many definitions in the field of informatics. From the literature review on context, an overview of its definitions is presented below:

- *Context represents a set of explicit variables that model contextual factors in the underlying domain (for example, time, place, surroundings, device, occasion, and so on). This definition comes from recommender systems (Adomavicius and Tuzhilin, 2011).*
- *Context is the set of environmental states and settings that either determines an application's behavior or in which an application event occurs and is interesting to the user. This definition is generally used in the field of context-aware computing (Chen and Kotz, 2000).*
- *Context can be considered to be everything that affects the computation except the explicit input and output. This definition stands from the perspective of computer programming (Lieberman and Selker, 2000).*
- *Context refers to the situation under which user's database access happens. This definition focuses on context-aware database support for ambient intelligence (Feng, Apers, and Jonker, 2004).*
- *Context a subjective concept that is defined by the entity that perceives it. It could be generally described as the subset of physical and conceptual states of interest to a particular entity. This definition emphasizes wearable computing systems (Pascoe, 1998).*
- *A context is the set of all context information characterizing the entities relevant for a specific task in their relevant aspects. Here, a contextual information is any information which can be used to characterize the state of an entity concerning a specific aspect; an entity¹¹ is a person, a place or in general an object; an aspect¹² is a classification, symbol- or value-range, whose subsets are a superset of all reachable states; a situation is the set of all known context information. This definition applies to the ubiquitous computing environments (Strang and Linnhoff-Popien, 2003).*

Besides, we retain a definition that seems to be a consensus on whatever the field of application: *context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves (Dey, 2001).*

¹¹An entity is relevant for a specific task, if its state is characterized at least concerning one relevant aspect (Strang and Linnhoff-Popien, 2003).

¹²An aspect is relevant, if the state with respect to this aspect is accessed during a specific task or the state has any kind of influence on the task (Strang and Linnhoff-Popien, 2003).

These definitions specify context as the characteristic information of an entity's circumstances. Such information that can be considered in an entity's context is called contextual information. It represents the values of contextual factors (Adomavicius and Tuzhilin, 2011). For example, for **hour** factor, the value **17h** is contextual information.

Many contextual factors are mentioned in the previous studies. According to these studies, some contextual factors are shown in Table 2.5. Note that these contextual factors are not related to the same entity. First three of them describe user context, while others separately focus on intelligent environment context (Franklin and Flaschbart, 1998), mobile device context (Rodden et al., 1998) and computer context (Ryan, Pascoe, and Morse, 1998).

TABLE 2.5: Summary of contextual factors.

References	Factors
Brown, Bovey, and Chen (1997)	user's location, who they are with, what the time of the day is
Palmisano, Tuzhilin, and Gorgoglione (2008)	user's intent and its changes over time
Rodden et al. (1998)	nearby devices, space and location
Franklin and Flaschbart (1998)	what is happening at this moment (current events)
Schilit, Adams, and Want (1994)	the location of use, the collection of nearby people, hosts, accessible devices, and their changes over time
Ryan, Pascoe, and Morse (1998)	location, time, temperature and user identity

TABLE 2.6: Classification of human-factors-related context (Schmidt, Beigl, and Gellersen, 1999).

Entity	Dimension	Factor
Human Factors	User	Knowledge of habits
		Emotional state
		Bio-physiological conditions
	Social Environment	Co-location of others
		Social interactions
		Group dynamics
	Task	Spontaneous activity
		Engaged tasks
		General goal

Grouping contextual factors makes it possible to specify a measurable dimension. For example, Schmidt, Beigl, and Gellersen (1999) classified human-factors-related context into three categories (dimensions): User, Social Environment and Task. Each dimension consists

of three contextual factors (see Table 2.6). It implies that an entity can use multiple contextual dimensions to identify its circumstances. In other words, an entity's context has the potential to become multidimensional.

In conclusion, the definition of an entity's context remains dependent on what it is intended to use. It is therefore necessary to identify the appropriate dimensions as well as the associated factors, which seeks to define the characteristic features of an entity. These features can be used not only to describe the entity at a given time but also to infer information of the future. For example, in forecasting weather, day d 's context is utilized to predict the weather for the next few days.

2.3.2 Classification

One easy approach to classify context is by **entities** (Dey, Abowd, and Salber, 2001) whose situation is described and characterized by contextual information. Zimmermann, Lorenz, and Oppermann (2007) concerned four entities types: natural, human, artificial and group (see Table 2.7). In the meantime, Dey, Abowd, and Salber (2001) identified three types: places, people and things¹³. Bazire and Brézillon (2005) only considered either an event (e.g., a collaboration) or an object.

TABLE 2.7: Classification of entities (Zimmermann, Lorenz, and Oppermann, 2007).

	Description
Natural entity	Natural entities indicate all living and non-living things that occur naturally and are not the result of any human activity or intervention (e.g., animals).
Human entity	Human entities refer to human beings (e.g., user).
Artificial entity	Artificial entities denote products or phenomena that result from human actions or technical processes (e.g., computer).
Group entity	Group entity is a collection of entities, which share certain characteristic, interact with one another or have established certain relations between each other (e.g., user group).

Under this classification, one of the most widely addressed context is **user context**, which mainly determines the behaviors of context-aware applications (Brown, Bovey, and Chen, 1997) and matters in predicting users' behaviors (Palmisano, Tuzhilin, and Gorgoglione, 2008). User context concentrates on the entity: user, which is also the entity of most works

¹³Things are either physical objects or software components and artifacts (e.g., an application or file) (Dey, Abowd, and Salber, 2001).

presented in Section 2.3. This is mainly because context is inevitably tied to an entity according to the definitions in Section 2.3.1 and in most cases the entity is user. However, whatever the entity, all works explore the notion **context (of an entity)**.

Depending on whether context changes over time or not, Adomavicius and Tuzhilin (2011) applied two categories to distinguish different contexts:

Static: Relevant contextual factors and their structure remains stable over time.

Dynamic: The contextual factors change in some way.

These contexts have a lifetime and their importance shifts over time. For example, during one month, a person's age is relatively static. However, if he/she moves to another city during the month, his/her address is dynamic. Therefore, from a perspective, the address is more important than the age for the given month.

Chen and Kotz (2000) identified two types of context with regard to its use in applications: **active** and **passive context**. Active context influences the behaviors of an application, while passive context is relevant but not critical to an application. For example, in a map application, user's location belongs to active context, while user's gender is passive context. Because the user's gender is not decisive in such an application. Instead, the application must change its behaviors when adapting to the user's location.

Another classification method lies in the difficulty of collecting contextual information. Through different levels for abstracting the entity's contextual information, context can be divided into **low-level context** and **high-level context** (Hong, Suh, and Kim, 2009). Low-level context comprises raw data collected directly from physical devices (e.g., sensors) (Hong, Suh, and Kim, 2009). As acquired without further interpretation, low-level context can be insignificant, trivial, vulnerable to small changes or uncertain (Ye et al., 2007). On the other hand, high-level context provides summary descriptions of one entity's state and surroundings (Wang et al., 2004b), derived from reasoning and interpreting low-level context (Bettini et al., 2010). For example, for a conversation between two persons, the low-level contextual factors include time, location, participants and conversation records, while its objective is a high-level factor that cannot be attained directly. Acquiring high-level context is already a challenge. But such information can be valuable and even have a determining impact on the entity.

Besides, context can also be separated into **explicit** and **implicit context** (Wang et al., 2004a) according to different collecting mechanisms. Explicit context is collected from context sources directly or required explicitly to the entity. As well, implicit context can either be additional information deduced from explicit context or be implicitly learned from the entity's activities or behaviors (Wang et al., 2004a; Anand and Mobasher, 2007). For example, on an online commercial site, user's comment on an item is explicitly required, which is

explicit context. But as implicit context, the frequency of the user's navigation clicks on the item is implicitly collected.

These multiple classifications of context don't conflict with each other. They are mutually compatible somehow. For instance, explicit context definitely belongs to low-level context as it is captured from sources or the entity itself without any processing. Meanwhile, implicit context can be either low-level or high-level.

Overall, the classification of context is not sufficiently stable. The main reasons are the complexity and reliance of context. Besides, due to the popularity of user context, we further investigate its contextual dimensions and associated factors by reviewing and analyzing previous works.

2.3.2.1 User context

Zimmermann, Lorenz, and Oppermann (2007) exploited five basic categories (dimensions) of an entity (see Table 2.8). An entity can refer to a user.

TABLE 2.8: Five contextual dimensions of an entity (Zimmermann, Lorenz, and Oppermann, 2007).

Dimension	Description
Individuality	Contains properties and attributes describing the entity itself
Time	Provides the time coordinates of the entity
Location	Provides the spatial coordinates of the entity
Activity	Covers all tasks that this entity is currently and in future involved in
Relation	Represents the information about any relationship that the entity has established to other entities

Another method to analyze dimensions of an entity's context was proposed by Dey, Abowd, and Salber (2001), which introduce four essential categories (dimensions) shown below. In their research, an entity can also be a user.

- **Identity** refers to the ability to assign a unique identifier to an entity.
- **Location** is expanded to include orientation and elevation, as well as all information that can be used to deduce spatial relations between entities, such as co-location, proximity, or containment.
- **Status (Activity)** identifies intrinsic characteristics of the entity that can be sensed.
- **Time** helps to describe a situation and makes it possible to leverage off the richness and value of historical information.

Sladic and Milosavljević (2017) applied seven dimensions to present context in business systems for access control requirements: Actor, Action, Resource, Means, Time, Place and Objective, which can be used to retrieve some current information from a system, like who is the current user.

Apart from the above, some methods are specific to user context. Kofod-Petersen and Cassens (2005) divided user context into five sub-categories (dimensions):

- **Environmental** dimension captures the users surroundings, such as things, services, people, and information accessed by the user.
- **Personal** dimension describes the mental and physical information about the user, such as mood, expertise and disabilities.
- **Social** dimension presents the social aspects of the user, such as information about the different roles a user can assume.
- **Task** dimension focuses on what the user is doing, it can describe the user's goals, tasks and activities.
- **Spatio-temporal** dimension is concerned with attributes like: time, location and the community present.

Negre (2018) retained 5 dimensions for user context in data warehouses: Time, Individuality/User Profile, Activity, Relation, Device. In addition, Ferdousi, Negre, and Colazzo (2017) proposed to analyze user context through 3 families and 10 dimensions:

- **Physical family** includes all dimensions on which the user's geographical position will have a strong influence.
 - **Temporal dimension** contains factors like week, season, time.
 - **Spatial dimension** is represented by two kind of factors: exact geographical position (e.g., longitude/latitude) and the nominal classes that can determine user's position (e.g., 'at home').
 - **Environmental dimension** expresses environmental characteristics, such as temperature and weather.
 - **Equipment dimension** describes all non-human objects that surrounds the user, such as printer.
- **Personal family** denotes the user's more specific information through four dimensions.
 - **Demographic dimension** consists of the user's identity information, such as name, age, and gender.
 - **Social dimension** refers to the appearances and roles of other people around the user and their relationships with the user.

- **Psycho-physiological dimension** reflects the user’s psycho-physiological aspect, such as the user’s mood and mental state.
- **Cognitive dimension** relates to the user’s experiences, constraints and activities.
- **Technical family** illustrates the characteristics of the devices utilized by the user.
 - **Hardware dimension** comprises the hardware to access the context-aware applications, such as processors and devices used by the user.
 - **Data dimension** focuses on the data manipulated by the applications, such as its format (text, video, audio, image).

Among all the different dimensions, three are common and included in all studies: **Time**, **Activity (Task)**¹⁴, and **Location**.

Besides, the dimension that contains information and attributes to describe **users (actors)** themselves is also discussed and explored as well. But this dimension was presented through various terms, such as individuality (Zimmermann, Lorenz, and Oppermann, 2007; Negre, 2018), identity (Dey, Abowd, and Salber, 2001), actor (Sladic and Milosavljević, 2017). Some studies even exploited two or more dimensions to present such contextual information of user: Ferdousi, Negre, and Colazzo (2017) introduced personal information by three dimensions (demographic, social, and psycho-physiological), while Kofod-Petersen and Cassens (2005) specified two (personal and social).

By comparison, Zimmermann, Lorenz, and Oppermann (2007) and Dey, Abowd, and Salber (2001) did not take into account the influence of **resources** used by users. Negre (2018) improved this by considering device. Sladic and Milosavljević (2017) and Ferdousi, Negre, and Colazzo (2017) proceeded further by proposing a dimension: resource (data). Kofod-Petersen and Cassens (2005) described it as information and services available to the user in environmental dimension. Such a dimension is significant to users, which indicates what the user is using. Due to the rapid development of information and communication technologies, digital resources and metadata must also be included. For example, hardware and equipment were taken into consideration by Ferdousi, Negre, and Colazzo (2017).

These diverse dimensions of user context reflect the complex dependencies of context, which raise challenges to context modeling. In order to present context properly and construct a corresponding model, we need to investigate and discuss existing approaches to model context.

¹⁴Only Ferdousi, Negre, and Colazzo (2017) represented it through cognitive dimension, while others defined it as activity/task dimension.

2.3.3 Modeling

A large number of context models were proposed with the development of context-aware applications. Based on the scheme of data structures, Strang and Linnhoff-Popien (2004) categorized six approaches of context modeling, as follows: Key-value model, Markup model, Graphical model, Object-oriented model, Logic-based model, and Ontology-based model. Moreover, Bettini et al. (2010) highlighted spatial models that concentrate on the location context. A context model may integrate multiple approaches (Strang and Linnhoff-Popien, 2004) to perform more flexibly (Bettini et al., 2010). Such hybrid models are listed in the most representative one.

2.3.3.1 Key-value model

As the most simple data structure of context modeling (Strang and Linnhoff-Popien, 2004), key-value model uses *key-value* pairs to define contextual factors and their values (Bettini et al., 2010). Based on Schilit, Adams, and Want (1994), the pair *location: in room GI136* can represent a user's location context. Such pairs are easy to manage, but lack sophisticated structuring capabilities to enable effective context retrieval algorithms (Strang and Linnhoff-Popien, 2004).

2.3.3.2 Markup model

Markup model depends on a hierarchical data structure (Strang and Linnhoff-Popien, 2004) that is describe by a variety of markup languages, such as XML (Bettini et al., 2010). It consists of mark up tags and content. Typical representatives of markup model are profiles (Strang and Linnhoff-Popien, 2004). For example, through a *Comprehensive Structured Context Profiles (CSCP)* proposed by Held, Buchholz, and Schill (2002), the user's location is presented in Figure 2.1. Markup model has scheme definitions and set of validation tools. However, it is limited in consistency and completeness checking. It is neither good at supporting reasoning on contextual information (Strang and Linnhoff-Popien, 2004; Bettini et al., 2010).

```
<?xml version="1.0" encoding="UTF-8"?>
<session:SessionProfile>
  <session:user>
    <session:location>
      In room GI136
    </session:location>
  </session:user>
</session:SessionProfile>
```

FIGURE 2.1: An example of the user's location represented by Markup model.

2.3.3.3 Graphical model

A well known language for this approach is the Unified Modeling Language (UML) which has a strong graphical component (UML diagrams) (Strang and Linnhoff-Popien, 2004). Then a representation of the user's location is shown in Figure 2.2. Graphical models have great advantages on the structure level. Nevertheless, they can't be formatted and interpreted by machines (Strang and Linnhoff-Popien, 2004).

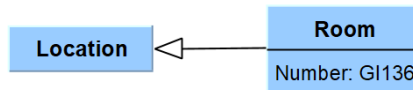


FIGURE 2.2: An example of the user's location represented by Graphical model.

2.3.3.4 Object-oriented model

Two main features of object-oriented models are encapsulation and reusability, which are also the main benefits of this approach (Strang and Linnhoff-Popien, 2004). Most spatial models follow this approach to organize their context information by physical location (Bettini et al., 2010). An example of the *Context Modeling Language (CML)* is in Figure 2.3, which was developed for conceptual modeling of databases but can also map from an object-oriented model to a runtime context management systems (Henricksen, Indulska, and Rakotonirainy, 2002; Bettini et al., 2010). This type of model contributes to solving problems arising from distributed compositions. And it can be implemented easily in a database. Unfortunately, it poses additional requirements (Strang and Linnhoff-Popien, 2004): it only processes the information formalized based on the model. That is, it can't be used to handle heterogeneous information from different sources and thus does not support interoperability (Bettini et al., 2010).

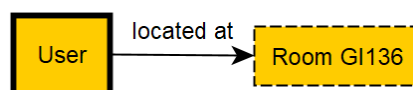


FIGURE 2.3: An example of the user's location represented by Object-oriented model.

2.3.3.5 Logic-based model

In a logic-based context model, the context is defined as facts, expressions, and rules. These rules form different conditions on which a concluding expression or fact may be reasoned or inferred from a set of other expressions or facts (Strang and Linnhoff-Popien, 2004). One of the first approaches was researched and proposed by McCarthy (1993), which focuses

more on context reasoning than modeling. This approach has extremely high level of formality and can be composed distributed. However, its partial validation is hard to maintain. A major issue is how it can be applied in computing environments.

2.3.3.6 Ontology-based model

Ontology provides a formal structure to manipulate knowledge related to a particular field or more broadly knowledge bases. An ontology-based model can be regarded as a formal and explicit specification of a shared, agreed and detailed conceptualization (Peckham and Maryanski, 1988; Gruber, 1993; Du et al., 2019; Munir and Anjum, 2018). Based on Du et al. (2019), Negre (2017), and Strang, Linnhoff-Popien, and Frank (2003), an ontology-based model includes two parts:

- 1) **T-Box** defines the nature of concepts (classes of subjects) and their interrelationships that constitute model (represented by the green rectangles in Figure 2.4).
- 2) **A-Box** explains the relationships between instances of the concepts and relations that are defined in T-Box (represented by the white rectangle in Figure 2.4).

An example is *Context Ontology Language (CoOL)* (Strang, Linnhoff-Popien, and Frank, 2003), which aims to enable context-awareness and contextual interoperability in a distributed system. An example about the user's location of this approach is shown in Figure 2.4.



FIGURE 2.4: An example of the user's location represented by Ontology-based model.

This approach offers flexibility and extensibility in distributed systems by supporting distributed storage and multi-author creation (Strang, Linnhoff-Popien, and Frank, 2003). It also provides a formal semantic presentation of knowledge in a distributed system. This indicates that ontology-based models are capable of processing heterogeneous information from different sources and supports interoperability. Besides, the available semantic representations (e.g., RDF, RDFS, OWL), which are used not only in authoring ontologies but also in automated information processing, enable ontology-based models to be machine-interpretable. Moreover, the two parts of an ontology-based model gives the opportunity for reusing the T-Box component in existing ontologies to construct new ones and extracting information from the A-Box component in existing ontologies to enrich new ones. Specifically, ontologies that are fully documented¹⁵ can provide useful input for ontology reuse and enrichment.

¹⁵https://www.w3.org/wiki/Good_Ontologies

All the advantages mentioned result in ontology being an appropriate choice among the six approaches. Therefore, regarding context as a specific type of knowledge (Bettini et al., 2010), ontology-based modeling is selected to represent context in this thesis.

2.3.4 Discussion

When talking about the notion **context**, it is always used and related to an entity. As the entity varies, so does its context, reflecting the reliance of the context. Moreover, the diverse properties of the context enable it to be classified in different manners, indicating its complexity. Thus, as a dependent and complicated notion, the definition of the context is still imprecise and somewhat unclear. This also generates difficulties for its representations. However, through identifying necessary contextual dimensions and factors, the context can be specified and represented, such as user context. This implies that these dimensions and factors can also contribute to developing a well-designed context model.

From the above analysis of the six context modeling approaches, the ontology-based approach is favored because of its flexibility, extensibility, interpretability, reusability, and its support for interoperability. Thus, this thesis focuses on presenting contexts through ontology-based models in the framework of collaborations.

Besides, as the most widespread context in informatics, user context is applied in many context-aware applications to provide more specific services and/or products for users. One is known as the context-aware recommender system, which will be discussed in Section 2.5.

2.4 Web-based collaborative working environment and system of information systems

As discussed in Section 2.2, collaboration matters for human activities. With the help of information technology, more and more people are collaborating remotely. This leads to the emergence of a new collaborative space: **collaborative working environment (CWE)** where people can work together as spontaneous and dynamic groups assembled in a collaborative manner (Prinz et al., 2006). CWEs, especially web-based CWEs, intend to support collaborations between users (i.e., people) by integrating and offering different collaborative tools. To explore such environments, their functionalities, frameworks, and features are studied in Section 2.4.1.

However, there are still unresolved problems in web-based CWEs, such as poor interoperability between heterogeneous tools and difficulties in managing distributed information (Carreras and Skarmeta, 2006; Hofte, 1998). Solving these issues necessitates to understand the relationship between a web-based CWE and its integrated tools. Each tool can be considered as an information system with its own information management (Neto, Araujo, and

Santos, 2017). These systems, together with the environment itself can form a **System of Information Systems (SoIS)**. To have a high-level understanding and viewpoint about such systems, we also investigate SoIS and its conceptual origins: **System of Systems (SoS)** by analyzing their definitions and features.

2.4.1 Web-based collaborative working environment

Collaborative Working Environment (CWE) is derived from the term **virtualized collaborative workplace** (Prinz et al., 2006) where eProfessionals can seamlessly collaborate to achieve common goals (Prinz et al., 2006; Martínez-Carreras et al., 2007). The notion **eProfessionals** extends the concept **professional** by including knowledge worker who intensively uses Information and Communication Technology (ICT) tools and services (Wang, 2016). These tools and services can support collaborative work, such as email, document sharing, and project management (Bafoutsou and Mentzas, 2002). And they are provided in CWEs (Truong et al., 2008) for assisting people to collaborate.

Another purpose of CWE is similar to those of **electronic workplace** (Bafoutsou and Mentzas, 2002) whose main objective is to provide groups with shared spaces to enable their members collaborate (Bafoutsou and Mentzas, 2002). Users can be members in different groups, and each group has a corresponding shared space allowing its members to collaborate (Bentley et al., 1997). By gluing different groupware applications (Martínez-Carreras et al., 2007), CWE can serve for users who collaborate in several groups at the same time.

Besides, another requirement raised for CWEs is: to facilitate the collaboration of a group geographically dispersed (Carreras and Skarmeta, 2006; Su et al., 2005; Kan, Duffy, and Su, 2001; Bafoutsou and Mentzas, 2002). In traditional collaborations, collaborators are not able to work together and exchange their ideas if they are situated in different locations (Kan, Duffy, and Su, 2001). To solve this issue, CWEs need to support e-collaborations (Kan, Duffy, and Su, 2001; Bafoutsou and Mentzas, 2002), which needs to use various technologies, such as Web/Internet technology, ICT, and technologies in Computer Supported Collaborative Work (CSCW) field¹⁶ (Martínez-Carreras et al., 2007; Su and Casamayor, 2009; Su et al., 2005). The development of these technologies gives people the opportunity to collaborate in a web-based CWE regardless of their geographical distances.

To analyze e-collaborations, Weiseth et al. (2006) defined a framework consisting of collaboration environment, process and support (see Figure 2.5). A collaboration environment can support various collaboration processes that should use different collaboration supports, such as organizational measures and collaborative tools. Meanwhile, these supports restricts the processes that are carried out in the environment.

¹⁶CSCW addresses how collaborative activities and their coordination can be supported by computer systems (Carstensen and Schmidt, 1999).



FIGURE 2.5: A collaboration framework (Weiseth et al., 2006).

As collaboration environments, web-based CWEs can influence collaboration processes and supports based on the framework in Figure 2.5. This, in turn, has an impact on the efficiency and success of collaboration. To help people succeed in their collaborations, web-based CWEs must be capable to supply collaborators with sufficient functionalities that are listed below and summarized from Martínez-Carreras et al. (2007), Neo (2003), Su and Casamayor (2009), Laso Ballesteros and Prinz (2008), Bafoutsou and Mentzas (2002), and Prinz et al. (2006):

- Allow people to collaborate over time and space
- Support collaborators' various activities during their collaborations, such as interactions with other users and/or resources
- Provide flexible services for users to support their collaborations
- Offer asynchronous collaboration tools¹⁷ (e.g., email and Wiki) and synchronous collaboration tools¹⁸ (e.g., real-time chat and video communication systems)
- Enable interoperability with different collaborative systems
- Increase the productivity and creativity in collaborative processes
- Enhance collaborators' critical and analytical thinking and problem-solving skills

Many researches already developed web-based CWEs to support collaborations in different domains. For example, Su et al. (2005) constructed a web-based CWE for online designers, where information is exchanged and stored in a database. Truong et al. (2008) aggregated disparate collaboration services (e.g., document sharing, communication, team management and project management services) and provide a web-based CWE, known as **inContext**, where runtime and historical context and users' interaction information are utilized to adapt services. Besides, Su and Casamayor (2009) applied a web-based CWE to enhance sustainable furniture design, which consists of three layers: upperware, middle ware and resources.

- **Upperware** interacts with different applications as collaborative tools to provide their corresponding services.

¹⁷Asynchronous collaboration tools allow users to collaborate at different times (Xu et al., 2008).

¹⁸Synchronous collaboration tools enable users to collaborate at the same time (Wang, 2016).

- **Middleware** manages the complexity and heterogeneity inherent in the distributed tools.
- **Resources** recognizes physical resources in the web-based CWE.

However, among these existing web-based CWEs, there are many difficulties that cannot be completely solved. For example, with the development of collaborative tools, interoperability between heterogeneous tools (Carreras and Skarmeta, 2006) is becoming increasingly difficult to achieve within a web-based CWE. It also poses problems to the common standard to share information between these tools (Martínez-Carreras et al., 2007). Switching between these tools during collaborations also creates problems to users, including (Hofte, 1998): 1) they might need to log into various tools for launching a particular collaboration activity; 2) they must copy and/or move information between tools within a web-based CWE when working with multiple tools. Particularly, such information is stored in different databases (either in the tools where it was produced or in the CWE itself) and applied in specific domains. This causes semantic gaps between information from different tools (Dou, Wang, and Liu, 2015), complicating both information access and management in a web-based CWE. Accordingly, it also raises challenges to how collaborative tools can be integrated into a web-based CWE (Prinz et al., 2006).

With the aim to address the existing problems and establish an improved web-based CWE that satisfies users' needs, it is essential to investigate and analyze several top-ranked features of future CWEs. Based on Martínez-Carreras et al. (2007) and Laso Ballesteros and Prinz (2008), these features include:

- 1) **Ease of use.** The easier the system is, the higher productivity and acceptance the CWE achieves.
- 2) **Interoperability.** Services offered by different collaborative applications should interoperate in order to facilitate aspects such as mobility, flexibility, and the use of different applications in a CWE.
- 3) **Scalability.** The CWE should have the ability to grow up according to the needs of the system and without losing the level of performance.
- 4) **Anyplace – Anytime.** It is pursued the pervasive collaboration, where collaborators can work together over time at anyplace.
- 5) **Low Cost of Entry.** The boundaries between different CWEs should be reduced in order to improve interoperability. Furthermore, this would facilitate the development of new CWEs as well as the use and integration of new tools.
- 6) **Locating required Information.** Users should be provided with accurate mechanisms to locate their needed information in CWEs. The retrieval of information in CWEs

should not be only oriented to get data but also to get knowledge, which necessitates appropriate information management and/or context management.

- 7) **Goal-oriented.** The services offered in a CWE should be oriented to solve different problems in an intelligent way, according to the pattern followed by a human being based on decomposing activities in goals.

Some feature mentioned above have been achieved, namely 1), 4), And 5). For example, web-based CWEs allow people to collaborate over time and space, satisfying the feature *any-place - anytime*. The boundaries between different web-based CWEs are reduced, achieving the feature *low cost of entry*. However, the other features still need to be developed and fulfilled, especially interoperability. The interoperability between collaborative tools can influence information access, management, and comprehension in web-based CWEs, reducing the efficiency of users' collaborations. Besides, the huge volume of information coming from different tools leads to information overload in web-based CWEs. This creates problems in processing and retrieving information based on users' needs, making it difficult for users to locate their required information.

We, therefore, need to improve information access and management in web-based CWEs so that information can be fetched to meet users' needs during their collaborations. To attain this, ontology can be applied in web-based CWEs, which can not only serve as a knowledge base for integrating and managing heterogeneous information, but also filter information to generate recommendations for users during their collaborations (Dou, Wang, and Liu, 2015; Ristoski and Paulheim, 2016; Dou, Wang, and Liu, 2015). For deploying and implementing ontology in a web-based CWE, we need to understand the relationship between the integrated tools and the web-based CWE.

2.4.2 From system of systems to system of information systems

The notion, **System of Systems** (SoS), arises from the need to more effectively implement and analyze large, complex, independent, heterogeneous systems working (or made to work) cooperatively (Azarnoush et al., 2006; Jamshidi, 2008). It indicates a set of independent and autonomous systems that are integrated together to accomplish common missions (Jamshidi, 2008; Karcianas and Hessami, 2011). As an inevitable term of systems, SoS receives a lot of interest.

However, there is no standard definition of SoS since it is still at its developing stages (Jamshidi, 2008). Consequently, we make a literature survey on the definitions of SoS, which is presented as follows:

Definition 1: A SoS is an assemblage of components which individually may be regarded as systems, and which possesses two additional properties: operational independence and managerial independence of the components (Maier, 1998).

Definition 2: A SoS is a set or arrangement of systems that results when independent and useful systems are integrated into a larger system that delivers unique capabilities (DoD, 2008).

Definition 3: SoS are large-scale integrated systems which are heterogeneous and independently operable on their own, but are networked together for a common goal (Jamshidi, 2008).

Definition 4: SoS is a new type of systems that is formed from the collaboration between its components, which by themselves are independent systems (Assaad, Talj, and Charara, 2016).

Definition 5: A SoS is a collection of dedicated systems that combine their resources and capabilities to create a new, more complex system that offers more functionality and performance than simply the sum of constituent systems (Ameur et al., 2017).

Definition 6: SoS are complex systems resulting from the interoperability of constituent systems, managing resources and capabilities with managerial and operational independence that collaborate to produce emergent behaviors to achieve a specified global mission (Mohsin et al., 2019).

All the definitions listed above consider SoS as an alliance of components that can be individually regarded as a system. In other words, a SoS can be separated into two parts: the **global system**, which is the result of the gathering of the components, and the **component systems**, which are independent and heterogeneous systems (Assaad, Talj, and Charara, 2016).

Component systems in a SoS are integrated together to achieve a common goal. The goal can be having higher performances (Jamshidi, 2008), achieving particular missions (Mohsin et al., 2019), saving cost, or offering more functionalities (Ameur et al., 2017). To some extent, SoS can be understood as the result from the collaboration of component systems (Mohsin et al., 2019). This implies the driving force behind SoS: to achieve higher capabilities than would be possible with a single component system (Azarnoush et al., 2006). Besides, SoS has no single point failure and continues to operate even in a dynamic environment (Assaad, Talj, and Charara, 2016).

Until now, many studies were undertaken to identify and analyze the characteristics of SoS, such as Assaad, Talj, and Charara (2016), Maier (1998), Gorod et al. (2007), and Di-Mario, Boardman, and Sauser (2009). Based on them, here is a summary of characteristics that distinguish SoS from other systems (Assaad, Talj, and Charara, 2016).

Component Systems

- **Autonomy** The autonomy of component systems encloses both operational and managerial independence. In other words, component systems operate independently not only in a SoS, but also if the SoS is disassembled. Component systems are separately acquired and integrated but maintain a continuing operational existence independent of SoS. Therefore, they can and will choose their decisions based on their interests.
- **Heterogeneity** Component systems are owned and managed by distinct organizations and stakeholders. That is, these components must be handled by different parties.
- **Belonging** Component systems can join or leave a SoS based on their choice. Sometimes this belonging will lose them part of their autonomy, but this loss will be compensated by the wins they will get from the SoS, which is the exact reason why they choose to belong.
- **Connectivity** Component systems have the capabilities to communicate and interoperate with each other in a SoS, which indicates component systems are connected together to enhance SoS capability.

Global System

- **Evolution** The autonomy of the components, means that they have the ability to evolve and change, regardless of the SoS. On the other hand, SoS work in an unpredicted, dynamic environment. All of that, in addition to the fact that a SoS's objectives change as well, leads to an ever evolving system, that must adapt to account for internal, together with external changes
- **Emergence** One of the most important characteristics of a SoS, and is inherited from complex systems. SoS achieves its main objectives through the integration of component systems and their interactions, which is not a property of any component systems but of SoS as a whole.
- **Diversity** SoS should offer a lot of functionalities. These diverse functionalities are not achieved in a single component system but in the whole SoS.

Beyond these characteristics, SoS can be classified into four categories: Directed, Acknowledged, Collaborative, and Virtual (Maier, 1998; Dahmann and Baldwin, 2008; Assaad, Talj, and Charara, 2016).

Directed: Directed SoS is built to fulfill specific purposes and centrally managed (e.g., the systems responsible of the development of the Future Combat Systems in the US Department of Defense).

Acknowledged: Acknowledged SoS has central management and common resources. Nevertheless, component systems retain their independent ownership and objectives.

Changes in the SoS are based on collaboration between SoS and component systems (e.g., Air Operation Center).

Collaborative: Collaborative SoS doesn't have central management with coercive power. Component systems collaborate to fulfill the central purposes (e.g., Intelligent Transport Systems).

Virtual: Virtual SoS lacks both central management and centrally agreed common purposes. It emerges from the interaction between components, whereas the purposes are unknown. This SoS is maintained through invisible mechanisms (e.g., national economies).

Particularly, if each component system in a SoS is an information system, this SoS can be viewed as a special type: System of Information Systems (SoIS) (Saleh and Abel, 2016). As a special SoS, SoIS always have the characteristics of SoS discussed before. And it can also be classified into the four categories (Directed, Acknowledged, Collaborative, and Virtual).

However, the specificity of SoIS lies in the difference between the information system and other systems. An information system contains a set of interrelated components perform activities aiming at collecting, processing, storing and distributing information, while a system is a set of elements dynamically interrelated to perform activities aiming at achieving a specific goal (Neto, Araujo, and Santos, 2017). In comparison, the characteristic of the information system is its objective concerning information. Thus, SoIS uses information from separated information systems to aggregate existing services and produce new ones. A good practice is the Internet. Internet is a SoIS that contains different information systems providing various services to users (see Figure 2.6).

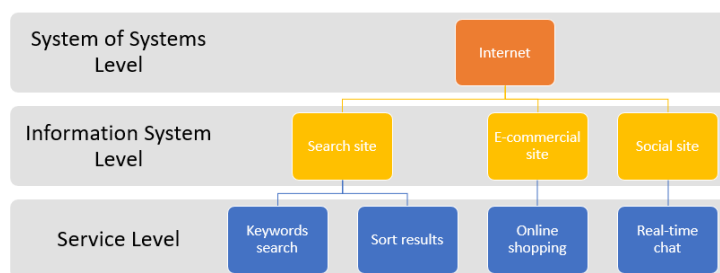


FIGURE 2.6: An example of SoIS: Internet.

To integrate new services, SoIS needs to concern the exchange of information and knowledge among different information systems. Therefore, the interoperability between the information systems is a key issue (Saleh and Abel, 2016). One method is to create a common language to describe data, where each information system can represent its data such that other systems may interpret (Bowen and Sahin, 2010).

Therefore, Saleh and Abel (Saleh and Abel, 2016) proposed applying ontology and a corresponding architecture of collaborative SoIS to establish a common knowledge base that can

interpret and manage information gathered from different information systems. This eases information communications between heterogeneous systems and thus achieves better interoperability in a SoIS. Except for ontology, an alternative common language is unavailable for SoIS to the best of our knowledge.

2.4.3 Discussion

The overviews of web-based CWE and SoIS reveal many common characteristics between the two notions. Both integrate heterogeneous collaborative tools or information systems in order to provide more advanced and flexible services to users. Besides, a common problem for both web-based CWE and SoIS is the interoperability between the integrated tools or systems.

However, there are also differences between web-based CWE and SoIS. Web-based CWE focuses more on achieving its purpose: to support users' collaborations. Insufficient attention is paid to how collaboration tools are integrated into a web-based CWE. SoIS gives a higher priority to investigating the characteristics of information systems and their relationships, whether they are component systems or the global system. Besides, the integrated systems of a SoIS are explicitly claimed to be autonomous, while those collaborative tools of a CWE are imprecise. This allows us to consider each tool in a web-based CWE as an autonomous information system with its own database. Thus, a web-based CWE with its integrated tools can be considered as a SoIS, precisely, a collaborative SoIS. While the web-based CWE refers to the global system, the tools are the component systems that work together to support collaborations.

Conceiving a web-based CWE as a collaborative SoIS can enhance its scalability and adaptation. This is because, as independent component systems, collaborative tools can be added into and/or removed from a web-based CWE, which satisfies the growing users' needs during collaborations without affecting the other integrated tools. Besides, the ontology-based SoIS and its architecture (Saleh and Abel, 2016) provide important guidance on how to deploy and implement ontology into web-based CWEs, thereby assisting in improving interoperability between collaborative tools and filtering information rapidly based on users' requirements within web-based CWEs.

2.5 Recommender systems

Recommender systems (RS) are information-filtering (advice-giving) systems that are designed and applied to find the information most relevant to users' needs and transfer it to users (Negre, 2015; Nunes and Jannach, 2017). In business field, many companies have already successfully used RS in their Web sites to help their customers, such as Amazon, Netflix, YouTube, iTunes, and Last.fm (Ricci, Rokach, and Shapira, 2011). Besides, RS is

an active research domain that contains many hot research topics and relates to data mining and machine learning techniques among others.

RSs gather various kinds of information to provide suggestions for users about items (Ricci, Rokach, and Shapira, 2011). Here, **item** is the general term to indicate the objects that RS recommends to users, while **user** indicates people who will receive these recommendations ((Ricci, Rokach, and Shapira, 2011)). For example, in a movie recommender system, users can receive recommendations on movies. An item refers to a movie. Usually, RSs only process two dimensions of information: User and Item. Such a RS is mentioned as **2D RS** in the rest of this thesis.

However, 2D recommendations are sometimes not quite relevant. Because user decision making, rather than being invariant, depends on the context when RSs provides recommendations (Adomavicius and Tuzhilin, 2011). For example, when recommending movies to users, RSs can incorporate information of users' companions to generate recommendations. To incorporate such contextual information into RSs, a new type of RS has been proposed and developed quickly in these years: Context-Aware Recommender system (**CARS**).

To distinguish these two RSs (2D RS and CARS) and clarify their specific features, an overview is provided for each type. It explores the functionalities, approaches, and techniques used in different RSs to generate recommendations.

2.5.1 2D recommender system

To provide recommendations, one core task of RSs is to identify and recommend useful items for users. This needs to predict which items are most suitable for users based on the given data. In 2D RSs, an item is characterized by its utilities (ratings) ((Ricci, Rokach, and Shapira, 2011)) that indicate how a particular user liked the item (Adomavicius and Tuzhilin, 2005). Then, the core task of 2D RSs is: **Given an initial set of ratings that users explicitly or implicitly give for items, 2D RSs try to calculate/compare items' unknown/missing ratings and decide which items to recommend, based on the data of User and Item** (Ricci, Rokach, and Shapira, 2011; Adomavicius and Tuzhilin, 2011).

Adomavicius and Tuzhilin (2005) modeled utility (rating) function of 2D RSs as follows:

$$R_{RS} : User \times Item \rightarrow Rating \quad (2.1)$$

where *Rating* is a totally ordered set (e.g., non-negative integers or real numbers within a certain range).

Most 2D RSs apply the utility (rating) function (presented by Equation 2.1) to generate recommendation of items to users, such as **Content-Based filtering (CB)** and **Collaborative Filtering (CF)** approach (Adomavicius and Tuzhilin, 2005; Koren, Bell, and Volinsky, 2009; Koren, Bell, and Volinsky, 2009; Adomavicius et al., 2005). However, some 2D RSs do

not use it. Instead, they apply heuristics to hypothesize that an item is useful to a user, which is typical in **Knowledge-Based (KB)** approach ((Ricci, Rokach, and Shapira, 2011)). Besides, another main approach used in 2D RSs is: **Hybrid** (Adomavicius and Tuzhilin, 2005; Balabanović and Shoham, 1997; Baltrunas et al., 2012; Panniello and Gorgoglione, 2011).

2.5.1.1 Content-Based filtering

Content-Based filtering (CB) approach creates a profile for each user and a description for each item to characterize their nature (Koren, Bell, and Volinsky, 2009; Adomavicius and Tuzhilin, 2005; Wang, 2016). By comparing the user profiles and item descriptions, 2D RSs can predict items' unknown utilities (ratings) for users (Adomavicius et al., 2005). Thus, CB approach can generally be deduced in two main steps shown below.

1. **Construct and learn item descriptions and user profiles:** Items are represented by a set of features, also called attributes or properties. When each item is described by the same set of attributes, and there is a known set of values the attributes may take, the description of each item is compiled with structured data (Lops, De Gemmis, and Semeraro, 2011). On the other hand, user profile is modeled with the same attributes and built up by analyzing the description of items to express users' preferences (Wang, 2016). It might also include demographic information or responses provided on a suitable questionnaire so that users can be associated with their preferred elements (Koren, Bell, and Volinsky, 2009). Usually, the techniques that are applied and implemented into this step involve Vector Space Model (VSM), TF-IDF, Semantic Analysis (using ontology) (Middleton, Shadbolt, and De Roure, 2004; Cantador, Bellogín, and Castells, 2008), Naïve Bayes, Decision Trees (Lops, De Gemmis, and Semeraro, 2011), Neural Networks (Oord, Dieleman, and Schrauwen, 2013). The first three techniques aim to construct descriptions and profiles, while Naïve Baye and Decision Trees are used to classify these descriptions and profiles.
2. **Calculate similarities between item descriptions and user profiles:** CB approach exploits the content of items to predict its relevance based on user profiles (Lops, De Gemmis, and Semeraro, 2011). In other words, it calculates the similarities between item descriptions and user profiles. This brings predictions of a user's interest in a particular item into a similarity problem. A widely used similarity in CB approach is Cosine similarity (Lops, De Gemmis, and Semeraro, 2011).

2D RSs applying CB approach recommend items that are similar to the ones that users liked in the past. The similarity is based on the attributes in descriptions of item and user profiles (Ricci, Rokach, and Shapira, 2011), which leads to dependence: the recommendations generated in CB rely on these predefined attributes. Thus, CB approach can provide

explanations on why the item is recommended to users based on these attributes and the calculated similarities (Ricci, Rokach, and Shapira, 2011). A possible explanation is that: "Item X was recommended because of its attributes A and B are shared by items Y and Z , which you liked" (Ricci, Rokach, and Shapira, 2011; Tintarev and Masthoff, 2007). However, such attributes sometimes require to gather the information that is not available or easy to collect (Koren, Bell, and Volinsky, 2009), which constitutes a limitation.

Another drawback is serendipity problem, which indicates that content-based 2D RSs tends to produce recommendations with a limited degree of novelty. That is, CB approach has no inherent method for finding something unexpected (Wang, 2016).

2.5.1.2 Collaborative Filtering

Collaborative filtering (CF) approach predicts an item's utility (rating) for a particular user based on users' past behaviors, such as previous transactions and items' utilities (ratings) (Koren, Bell, and Volinsky, 2009; Adomavicius et al., 2005). The general assumption of CF is: if two users have same utility (rating) on one item, one of them is more likely to have the similar utility (rating) as the other on a different item (Wang, 2016). Under this, two primary areas of CF have been developed, known separately as memory-based (neighborhood) and model-based CF (Koren, Bell, and Volinsky, 2009; Adomavicius et al., 2005; Koren and Bell, 2015; Breese, Heckerman, and Kadie, 1998; Su and Khoshgoftaar, 2009).

Memory-based (neighborhood) CF concentrates on the similarities between items or, alternatively, between users (Koren, Bell, and Volinsky, 2009). Calculating the similarities can be done in two ways: user-based and item-based recommendation (Desrosiers and Karypis, 2011). To predict a given user u 's rating for a particular item j , user-based recommendations are generated by evaluating other 'similar' users' ratings for item j , while item-based recommendations rely on user u 's ratings for other 'similar' items (Desrosiers and Karypis, 2011). All these similarities are calculated from users' past ratings using different techniques, such as Nearest neighbor (Sarwar et al., 2001), Pearson correlation (Jin, Chai, and Si, 2004), Cosine similarity (Desrosiers and Karypis, 2011) and similarities in ontology (Zhang, Gong, and Xie, 2013). All these techniques focus on finding 'similar' items or users.

Model-based CF utilizes the pure rating data (Su and Khoshgoftaar, 2009) to estimate or learn a model to make predictions (Breese, Heckerman, and Kadie, 1998). For predicting a given user u 's rating for a particular item j , Bayesian networks apply other users' ratings of the item j (Breese, Heckerman, and Kadie, 1998), while Markov decision process handles the user u 's last several ratings for items (Shani, Heckerman, and Brafman, 2005). As for latent factor models, they transform both items and users to a same latent factor space where both items and users are characterized through factors inferred from utilities (ratings) to explain users' ratings (Koren, Bell, and Volinsky, 2009; Koren and Bell, 2015). Some

techniques used in latent factor models are Matrix Factorization (MF), Latent Dirichlet Allocation (LDA), Singular Value Decomposition (SVD) (Koren, Bell, and Volinsky, 2009; Koren and Bell, 2015; Billsus and Pazzani, 1998; Su and Khoshgoftaar, 2009).

CF approach can provide accurate recommendations without any need for exogenous information about neither items nor users (Koren and Bell, 2015). Often, 2D RSs in CF approach can deal with very large data sets (Wang, 2016), such as the data set in Netflix competition including 17,770 movies rated by over 480,000 users (Koren and Bell, 2015). Besides, another advantage of CF is domain free, which indicates CF can address data aspects that are often elusive and difficult to profile using CB (Koren, Bell, and Volinsky, 2009). CF may apply users' implicit feedback (Koren, Bell, and Volinsky, 2009), such as users' click-throughs on items. Besides, CF approaches are able to give explanations to users, such as: *Other users similar to you liked the item X* (Ricci, Rokach, and Shapira, 2011), *The item X was recommended because people who liked Y also liked X* (Tintarev and Masthoff, 2007).

However, CF approach faces two main challenges: sparsity (Desrosiers and Karypis, 2011) and cold-start (Koren, Bell, and Volinsky, 2009). The sparsity problem is quite common due to the fact that users typically rate only a small proportion of the available items (Desrosiers and Karypis, 2011). When a new item or user is added, there is not any data from old users' past behaviors to generate recommendations. Sometimes, sparse rating data problem can aggravate and be hard to solve, known as cold-start (Koren, Bell, and Volinsky, 2009; Desrosiers and Karypis, 2011). These challenges to CF approaches are mainly because it is unlikely that two users or elements have common ratings when the rating data is sparse (Desrosiers and Karypis, 2011). This contrasts with the general assumption¹⁹ of CF approaches, which impacts their performance in 2D RSs.

2.5.1.3 Knowledge-Based

Knowledge-based (KB) approach generates recommendations using specific domain knowledge about users' needs and preferences. There are two basic specifics of KB approach: case-based (Bridge et al., 2005; Burke, 2000; Lorenzi and Ricci, 2003) and constraint-based (Felfernig et al., 2006; Felfernig and Burke, 2008; Felfernig et al., 2011). Both approaches are using knowledge bases (e.g., ontologies (Chen et al., 2012; Tarus, Niu, and Mustafa, 2018)) and collecting users' requirements to generate recommendations (Felfernig et al., 2011).

Case-based methods determine recommendations by evaluating similarity metrics (Felfernig et al., 2011) between the predefined cases and the user's requirement. The predefined cases are included in a case base (Bridge et al., 2005), which refers to the knowledge base of items. From a case-based viewpoint, the item description are cases (Bridge et al., 2005). This

¹⁹The general assumption is introduced in Page 39: if two users have same opinions on one item, one of them is more likely to have the similar opinions as the other on a different item (Wang, 2016).

implies that the similarity metrics examines which item can best match the user's requirement according to a knowledge base predefined. Thus, recommendation problems are solved by measuring how closely items come to satisfying the user's requirement (Burke, 2000).

As for constraint-based methods, explicit rules are necessary. The rules are about how to relate users' requirements with item features (Felfernig et al., 2011). Therefore, the knowledge base contains item descriptions and explicit rules. Generating recommendations then turns into a constraint satisfaction problem (Felfernig and Burke, 2008), where the recommended item should satisfy certain rules according to users' requirements.

KB approach estimates the extent to which an item can meet users' explicit requirements based on a predefined knowledge base. Specifically, users' requirements are directly elicited within a recommendation session (Felfernig et al., 2011). This indicates that all the necessary data for generating KB recommendations can be extracted or predefined. Thus, there is no sparsity or cold start problem with KB approach (Felfernig et al., 2011). Besides, KB approach can produce explanations to users, such as: *The item X has advantages over the previous recommended item Y in features A and B, which will make X more appropriate as you requested* (Ricci, Rokach, and Shapira, 2011).

However, KB RSs have one drawback: knowledge acquisition bottleneck, which motivates knowledge engineers to convert the knowledge possessed by domain experts into formal, executable representations (Felfernig et al., 2011).

2.5.1.4 Hybrid

Hybrid approach combines two or more other native approaches (CB, CF, and KB) (Adomavicius and Tuzhilin, 2005; Baltrunas et al., 2012; Panniello and Gorgoglione, 2011; Burke, 2002). A hybrid 2D RS can be classified into three categories: monolithic, parallelized, and pipelined hybrids. Based on Jannach et al. (2010) and Burke (2002), different hybridization techniques are assigned to the three categories and illustrated as follows.

1. **Monolithic hybrids** consist of a hybrid recommender component that integrates multiple native approaches by preprocessing and combining several different data sources.
 - *Feature combination*: This technique is used to simply combine and preprocess several types of data sources. For example, when merging CB and CF, collaborative information is processed as additional feature data associated with each item and use CB techniques over this combined data set. However, the combination of input features using KB approaches with CB or CF sources has remained largely unexplored.
 - *Feature augmentation*: This technique applies more complex transformation steps, compared to feature combination. So that the output of the hybrid recommender

component augments the feature space of the applied native approaches by pre-processing their knowledge sources. Specifically, such a hybrid recommender is strongly linked to the main components in 2D RSs for performance and functionality reasons, such as a pseudo-user-rating (Melville, Mooney, and Nagarajan, 2002).

2. **Parallelized hybrids** employ several recommender components side by side and apply a specific hybridization mechanism to aggregate their outputs.

- *Weighted*: This technique computes weighted sums of ratings for items based on the rating results obtained from all available native approaches in a hybrid 2D RS.
- *Switching*: This technique uses some criteria to switch between different native approaches. Thus, it requires an oracle that decides which native approach should be used in a specific situation, depending on the user profile and/or the quality of recommendation results.
- *Mixed*: This technique is practical to make large number of recommendations simultaneously, where recommendations from more than one native approach are presented together at the level of the user interface.

3. **Pipelined hybrids** implement a staged process in which several native approaches sequentially build on each other before the final one generates recommendations for users.

- *Cascade*: This technique is based on a sequenced order of recommender components, in which each succeeding recommender only refines the recommendations of its predecessor.
- *Meta-level*: In this technique, one recommender component builds a model that is exploited by the principal recommender to make recommendations. For instance, Balabanović and Shoham (1997) exploited a principal CF recommender, based on user models that are built by a CB recommender.

Hybrid approach aims at overcoming the limitations of native approach (CB, CF, and KB) and improving the prediction performance. It has already been used in many 2D RSs, such as Google news RS (Das et al., 2007). However, no single hybridization technique is applicable in all circumstances (Jannach et al., 2010). This implies that their performances differentiates among different data sources and recommendation problems. Thus, it is necessary to analyze the advantages and disadvantages of these hybridization techniques. **Monolithic hybrids** are valuable if little additional knowledge is available for inclusion in the data source; As an additional post-processing step, **parallelized hybrids** are the least invasive for existing native approaches, but they add runtime and matching of ratings generated by these approaches; **Pipeline hybrids** require a deeper understanding of the algorithm's functioning to ensure

efficient runtime computations, but work well in some cases, such as when CF and KB are combined (Jannach et al., 2010).

2.5.2 Context-aware recommender system

Recently, a new branch of RS, **Context-Aware Recommender System (CARS)**, was proposed by Adomavicius et al. (2005). A CARS is constructed by incorporating context into 2D recommendation generating process, which can result in more accurate recommendations (Palmisano, Tuzhilin, and Gorgoglione, 2008). Thus, CARS is attracting more and more attentions in both business and academic field. Sourcetone interactive radio (www.sourcetone.com) started to consider the current mood of listeners (the context) for recommending songs in a CARS (Adomavicius and Tuzhilin, 2011). Besides, CARS has often been a research topic in the conference RecSys since 2009²⁰.

Unlike 2D RS, CARS deals with at least three dimensions of data: User, Item and Context. Sometimes, it is even possible to construct a CARS using over three dimensions of data. For example, Adomavicius et al. (2005) handled data separated in 5 dimensions: User, Item, Place, Time and Companion in a context-aware movie RS. Thus, the rating function of a n -dimensional CARS is (Adomavicius et al., 2005):

$$R_{CARS} : D_1 \times D_2 \times \dots \times D_n \rightarrow Rating(n \geq 3, n \in \mathbb{N}^*) \quad (2.2)$$

where D_1, D_2, \dots, D_n represent n dimensions of CARS (including User, Item, Context, ...). Particularly when $n = 3$, the rating function is: $User \times Item \times Context \rightarrow Rating$.

Since various contextual information is involved in CARS, Adomavicius and Tuzhilin (2011) identified two critical properties of CARS: complexity and interactivity. Here, complexity indicates that CARS can be significantly more complex in comparison to 2D RSs; interactivity implies that contextual information usually needs to be retrieved from users in CARS.

To integrate context into a CARS, two main approaches are proposed (Adomavicius and Tuzhilin, 2011):

1. **Context-driven querying and search:** This approach uses and specifies context in queries to search for the most appropriate item. Here, the queries are used to extract relevant rating data based on the implicit or explicit contextual information. For example, Google Map permits users to search for nearby locations, where a data query filters out irrelevant locations using the current position of users (user context).

²⁰<https://recsys.acm.org/recsys09/call/>

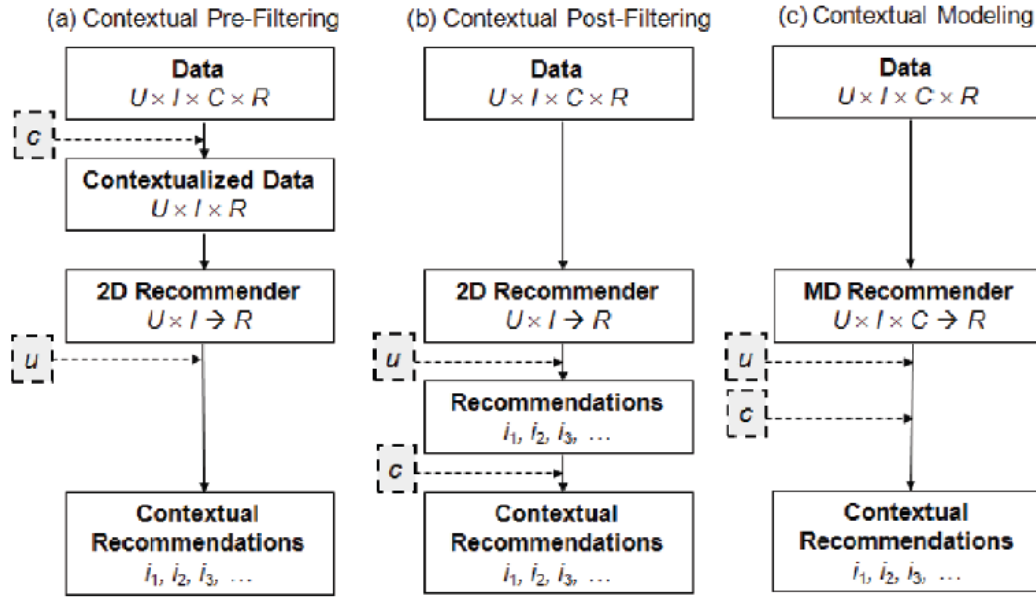


FIGURE 2.7: The incorporation of context in the recommendation process (Adomavicius and Tuzhilin, 2011).

2. **Contextual preference elicitation and estimation:** This approach applies context for modeling and learning user preferences to predict unknown ratings of items. For example, Liu, Xie, and Chen (2018) modeled researchers' preferences in collaborations that are presented by several indices: topics' similarity, ratio of new collaborator, researchers' conservativeness and activeness, which all belong to user context.

Based on the two approaches, three methods (see Figure 2.7) are proposed to incorporate context in different phases of recommendation processes (Adomavicius and Tuzhilin, 2011): **contextual pre-filtering (PreF)**, **contextual post-filtering (PoF)** and **contextual modeling (CM)**. All three methods can be adapted to the approach: contextual preference elicitation and estimation, while only contextual pre-filtering method is available for the approach: context-driven querying and search.

2.5.2.1 Contextual pre-filtering

First, contextual pre-filtering method (PreF) (shown in Figure 2.7 (a)) applies contextual information to filter utilities (ratings) that are irrelevant to specific contexts (Adomavicius et al., 2005). Particularly, context can essentially serve as a query for selecting relevant utilities (ratings) (Adomavicius and Tuzhilin, 2011). If the data filtering query is constructed using exactly the specified context (e.g., movies made in December 2019), it is exact PreF (Adomavicius and Tuzhilin, 2011). Alternatively, generalized PreF applies the query that refers to some generalization of the specified context (e.g., movies made in winter 2019) (Adomavicius et al., 2005).

Afterwards, approaches of 2D RSs (presented in Section 2.5.1) are used to predict unknown utilities (ratings) and thus generate recommendations for users. This results in one major advantage of this method: PreF supports the use of any 2D recommendation approach and technique (Adomavicius and Tuzhilin, 2011).

2.5.2.2 Contextual post-filtering

The basic idea of contextual post-filtering (PoF) (shown in Figure 2.7 (b)) is to analyze the contextual data for a given user in a particular context to find usage patterns of specific items, and then use these patterns to adjust the item list, resulting in context-aware recommendations (Adomavicius and Tuzhilin, 2011). In other words, the first step is applying approaches of 2D RSs (presented in Section 2.5.1) to have un-contextual recommendation results. Thus, an advantage of this method lies in the capacity to use any 2D recommendation approach and technique.

Then, the particular context is applied to filter out irrelevant results or adjust the ranking of recommendations. Two different techniques are available (Adomavicius and Tuzhilin, 2011).

- **Heuristic post-filtering** focuses on finding common item attributes (characteristics) for a given user in a particular context and then use these attributes to adjust the recommendations. For example, an attribute can be an actor (actress) that the user prefer to watch his (her) movies in a particular context.
- **Model-based post-filtering** can build predictive models that calculate the probability with which the user chooses a certain type of item in a particular context and then use this probability to adjust the recommendations (e.g., probability of relevance that males choose to watch a romantic movie in a particular context). Then, the probability is used to adjust the recommendations. Particularly, Panniello and Gorgoglione (2012) proposed two ways to adjust the recommendations: weight PoF reorders the recommendations by weighting the predicted rating with the probability, while filter PoF filters out recommendations that have small probability (Panniello and Gorgoglione, 2012; Adomavicius and Tuzhilin, 2011).

2.5.2.3 Contextual modeling

Contextual modeling (CM) methods (shown in Figure 2.7 (c)) directly use context inside the recommendation generation process as explicit attributes of a user's rating for an item. In other words, approaches of 2D RSs (presented in Section 2.5.1) can no longer be applied to implement CM without any changes.

CM gives rise to truly multidimensional recommendation functions, which essentially represents predictive models or heuristic calculations that incorporate contextual information in addition to the user and item data (Adomavicius and Tuzhilin, 2011), i.e., $Rating = R_{CARS}(D_1, D_2, \dots, D_n) (n \geq 3, n \in \mathbb{N}^*)$ ²¹.

Adomavicius et al. (2005) presented a CM method, known as *contextual-neighbors CM*. It defines a contextual profile $Prof(i, c)$ for each user, which aims to calculate cosine similarities among users and to find N nearest neighbors of user i in a specific context c .

2.5.3 Discussion

From previous research, the techniques and methods of CARS are still immature when compared to those of 2D RS. However, there is something in common between CARS and 2D RS approaches. For example, both monolithic hybrids and PreF must pre-process the data. Parallelized hybrids and PoF need a post-process step in the recommendation generating process. This indicates that 2D RS approaches are important sources of inspiration for CARS approaches, thus contributing to their development. Accordingly, methods and techniques in 2D RSs can be borrowed and adapted to generate context-aware collaborator recommendations in CWEs.

Specifically, we summarize the 2D recommendation approaches that can be directly used in a CARS method (see Table 2.9). Although both PreF and PoF can support any 2D recommendation approaches, they require extra steps in generating recommendation: PreF needs to first filter out irrelevant data from sources, as pre-processing step, and PoF must filter out irrelevant 2D recommendation results, as post-processing step (as shown in Figure 2.7). Besides, CM necessitates more complicated rating functions to deal with three or more dimensions of data in the recommendation generating process (as shown in Figure 2.7). Consequently, any 2D recommendation approaches can not be directly applied in CM.

TABLE 2.9: Summary of 2D recommendation approaches that can be directly used in CARS methods.

	CB	CF	KB	Hybrid
PreF	Yes	Yes	Yes	Yes
PoF	Yes	Yes	Yes	Yes
CM	No	No	No	No

Moreover, the multidimensional data in CARS also causes difficulties: from the perspective of data volume, CM can not handle the same amount of users and items as the other two methods (PreF and PoF). While CM needs to process data of $D_1 \times D_2 \times \dots \times D_n$, the other two focus only on data of $D_1 \times D_2$. Meanwhile, as the dimensions of the data increase, the

²¹Here, D_1, D_2, \dots, D_n represent n dimensions of CARS (including User, Item, Context, ...).

computational complexity of CM becomes higher and higher. However, for the other two methods (PreF and PoF), only their extra steps (pre-processing and post-processing) become more complicated. This indicates that, unlike PreF and PoF, CM is too costly when data volume and complexity are heavy. Thus, we concentrate on applying PreF and PoF to generate context-aware collaborator recommendations in this thesis.

As discussed in Section 2.5.1, each 2D recommendation approach (CB, CF, KB and Hybrid) has its own advantages and disadvantages in terms of data entry and recommendation outcomes. For instance, KB always places higher requirements than the others due to its use of knowledge bases, making it non-universal and high cost. This necessitates us to consider and adopt these approaches following the needs of the context-aware collaborator recommendations in this thesis.

Specifically, all these 2D recommendation approaches employ a semantic technique: ontology. In CB approach, ontology is used to construct user profiles and/or item descriptions (Middleton, Shadbolt, and De Roure, 2004; Cantador, Bellogín, and Castells, 2008); within CF, ontology serve to calculate similarities between items or users (Zhang, Gong, and Xie, 2013); in KB, ontology is applied to build knowledge bases (Chen et al., 2012; Tarus, Niu, and Mustafa, 2018); ontology has various possibilities of use due to the combined native approaches (CB, CF, and KB) in Hybrid. This enables us to work differently with the ontology depending on the needs of 2D RS, indicating that ontology is a flexible and extensible technique in generating 2D recommendations. Besides, as a semantic representation, ontology can also be combined with data mining techniques to explore semantic data (Ristoski and Paulheim, 2016; Dou, Wang, and Liu, 2015). It helps narrow the semantic gaps between data from different domains and/or sources (Dou, Wang, and Liu, 2015), enriching data entry in 2D RSs. We are therefore interested in implementing ontology to generate context-aware collaborator recommendations.

2.6 Chapter summary

In this chapter, we first introduced collaboration and presented the various factors that can affect its success. Analyzing and adjusting the integration of these factors can make collaborations have a greater chance of success. Particularly, the factor **context** is so complex that it may influence other factors as well. It also has an impact on both the actual process and the effectiveness of collaboration (Patel, Pettitt, and Wilson, 2012). This necessitates considering these factors within context rather than merely integrating them together. Besides, recommending collaborators also assists in achieving successful collaborations.

Then a retrospect about context was presented. Here, we explored definitions of context and emphasized three basic elements in constructing an entity's context: *contextual information*, *factors*, and *dimensions*. Based on previous work, the definitions of context are not uniform. This poses challenges in defining the context of the collaboration and building its

model. Thus, multiple approaches of context modeling were also presented. After comparing these approaches, we notice the advantages of the ontology-based modeling approach in terms of flexibility, extensibility, interpretability, reusability, and its support for interoperability between multiple information systems. Hence, we choose to construct an ontology-based collaboration context model (i.e., a collaboration context ontology).

Afterward, two notions, web-based Collaborative Working Environment and System of Information Systems, were investigated and compared. The similarity between them inspires us that a web-based CWE is a collaborative SoIS when its integrated tools are independent. Such a relationship implies that the solutions provided for collaborative SoISs can also be transferred to web-based CWEs to handle similar problems. Therefore, the architecture of ontology-based collaborative SoISs (Saleh and Abel, 2016) can also be considered from the viewpoint of web-based CWEs, providing an opportunity to apply and integrate the collaboration context ontology into web-based CWEs. As for the appropriate collaborators, establishing a context-aware collaborator recommender system in web-based CWEs is a suitable option.

Finally, 2D recommender system and context-aware recommender system were distinguished, including the approaches and techniques applied in these recommender systems. We also analyzed their advantages and disadvantages in generating recommendations. After excluding CM, appropriate 2D recommendation approaches should be implemented in PreF and PoF to generate context-aware collaborator recommendations. Additionally, due to the flexibility and extensibility of ontology in generating 2D recommendations, we therefore have interests in employing the collaboration context ontology for recommending collaborators to users in web-based CWEs.

In the next chapter, we will present the MEMORAE approach, which assists us in defining and implementing the collaboration context ontology into web-based CWEs.

Chapter 3

A survey of MEMORAe approach

3.1 Introduction

The MEMORAe approach encompasses a web platform and a core ontology, originally intended to manage resources within organizations and to support the organizational learning process (Atrash, Abel, and Moulin, 2014). With its developments, the MEMORAe approach has been used for other purposes. Specifically, it has been once employed to support collaborations in organizations (Deparis, 2013; Wang, 2016) and to manage resources in a System of Information Systems (SoIS) (Saleh and Abel, 2016). Both are close to our focus in this thesis: supporting users's collaborations in web-based CWEs which can also be considered as collaborative SoISs. This signifies that it is necessary to explore the MEMORAe approach.

Thus, the remaining parts of this chapter are organized as follows: Section 3.2 introduces a continuation of the MEMORAe approach, which can facilitate individuals' collaborations through a core ontology *MC2* and a web platform *E-MEMORAe2.0* (Atrash, 2015; Deparis, 2013). Section 3.3 presents another continuation, including a core ontology *SOIS* and a web platform *MEMORAe SOIS*, aiming at resource management in a SoIS (Saleh and Abel, 2016). Next, we discuss the advantages and disadvantages of the MEMORAe approach in Section 3.4. At the end of this chapter comes the conclusion.

3.2 MC2 and E-MEMORAe2.0

As an ontology, *MC2* concentrates on three main modules: **Individuals and groups of individuals**, **Resource**, and **Activity**, by integrating with four existing ontologies: SIOC¹, FOAF², PROV³, and VCard⁴ (Deparis, 2013; Atrash, 2015). Its general view is shown in Figure 3.1. All concepts in *MC2* starts with the prex *mc2* indicating its namespace.⁵

¹<https://www.w3.org/Submission/sioc-spec/>

²<http://xmlns.com/foaf/spec/>

³<https://www.w3.org/TR/prov-overview/>

⁴<https://www.w3.org/TR/vcard-rdf/>

⁵The complete *MC2* ontology is available at <https://gitlab.utc.fr/lisiying/ontologies-in-the-thesis-of-siyang-li.git>.

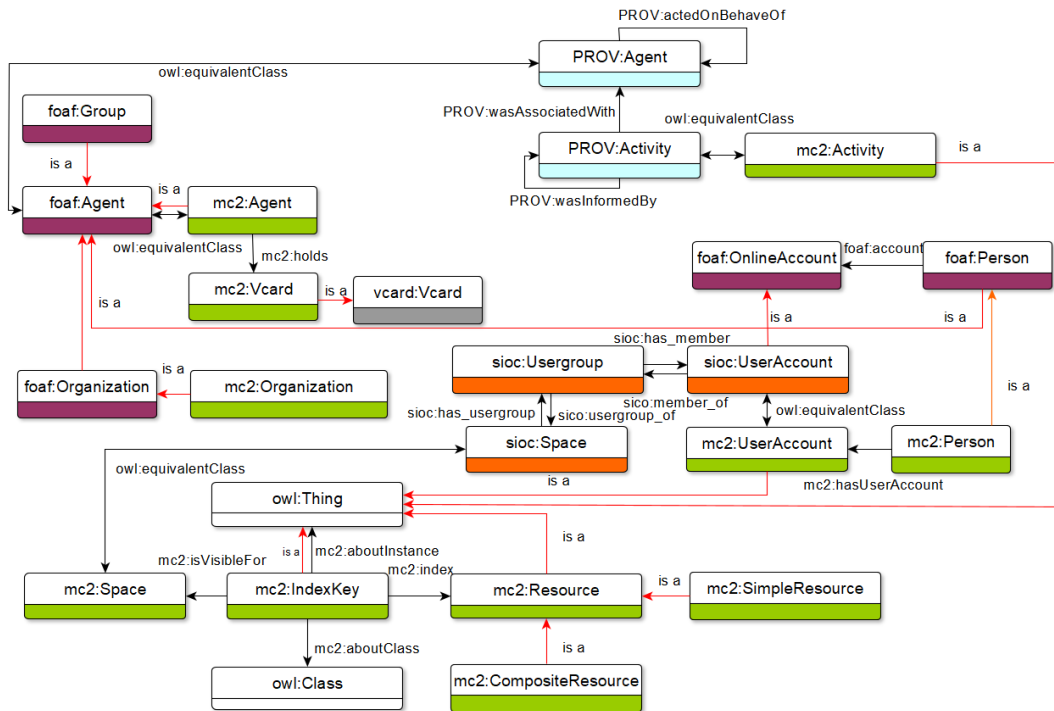


FIGURE 3.1: General view of MC2 with its three main modules (Deparis, 2013; Atrash, 2015).

Individuals and groups of individuals

This module is instantiated from SIOC and FOAF ontology (see Figure 3.2). An individual is a person that may have one or more user accounts. These user accounts can be members of groups, either personal, institutional, or free. Specifically, each group has its own space where its members carry out activities and operate on different resources.

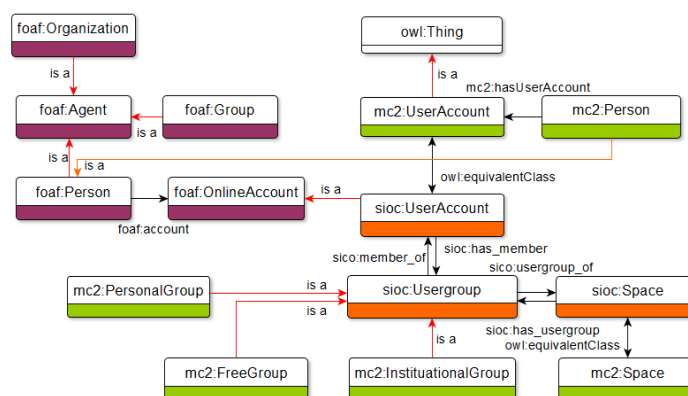


FIGURE 3.2: Individual and group module of MC2 (Deparis, 2013).

Each individual holds a VCard (see Figure 3.3). A VCard⁶ represent a virtual contact file for individuals, where their personal information (e.g., address, telephone number) is contained.

⁶<https://en.wikipedia.org/wiki/VCard>

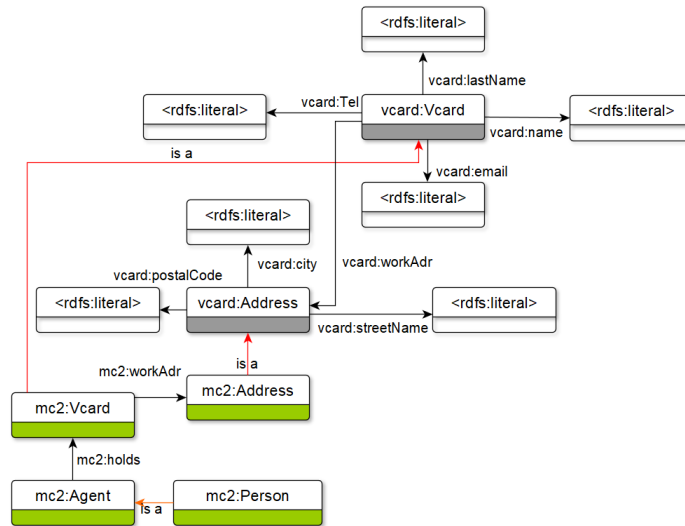


FIGURE 3.3: VCard in MC2 (Atrash, 2015).

Resource

MC2 divides resources into two categories: simple and composite (see Figure 3.4). A document, a vote or an agent can be instances of simple resources. An agent is something that does stuff, such as a person, group or organization. Composite resources are more complex, usually containing other resources.

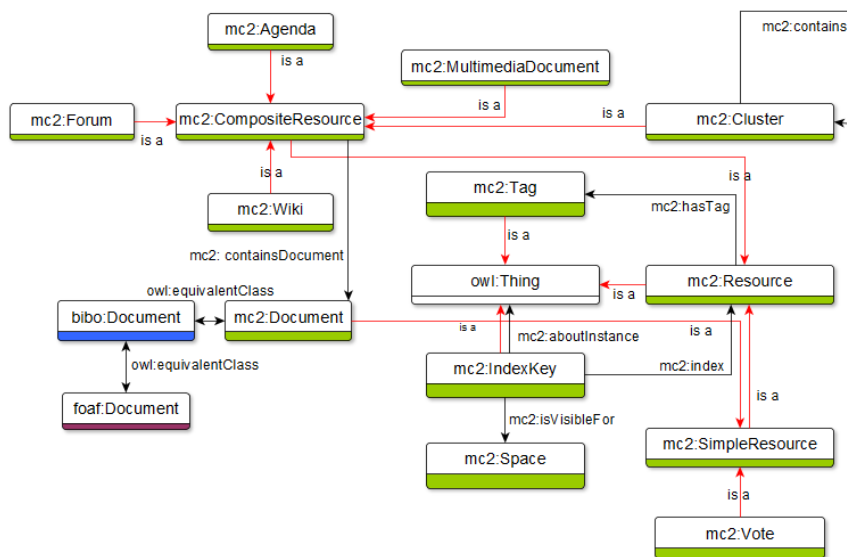


FIGURE 3.4: Resource module of MC2 (Deparis, 2013).

Each resource is referenced by an index key that is visible at a group space. This ties a resource to a group. Besides, one resource is also associated with a concept in a semantic map, which enables users to browse and interact with resources available in different groups.

Activity

Activity module of MC2 represents the processes and procedures done over time. It defined two types of activities: *mc2:ProceduralActivity* and *mc2:InteractionActivity* (see Figure 3.5). Specifically, the second represents interaction activities in the web platform through importing PROV ontology and identifying six sub-activity types: creating, deleting, modifying, accessing, adding, and sharing (Wang, 2016). These activities focus on individuals' interactions with resources. Each interaction activity takes place over a time period and acts on or with resources. This can compose and generate individual activity traces. Each trace belongs to an individual and is linked to the index key of the concerned resource.

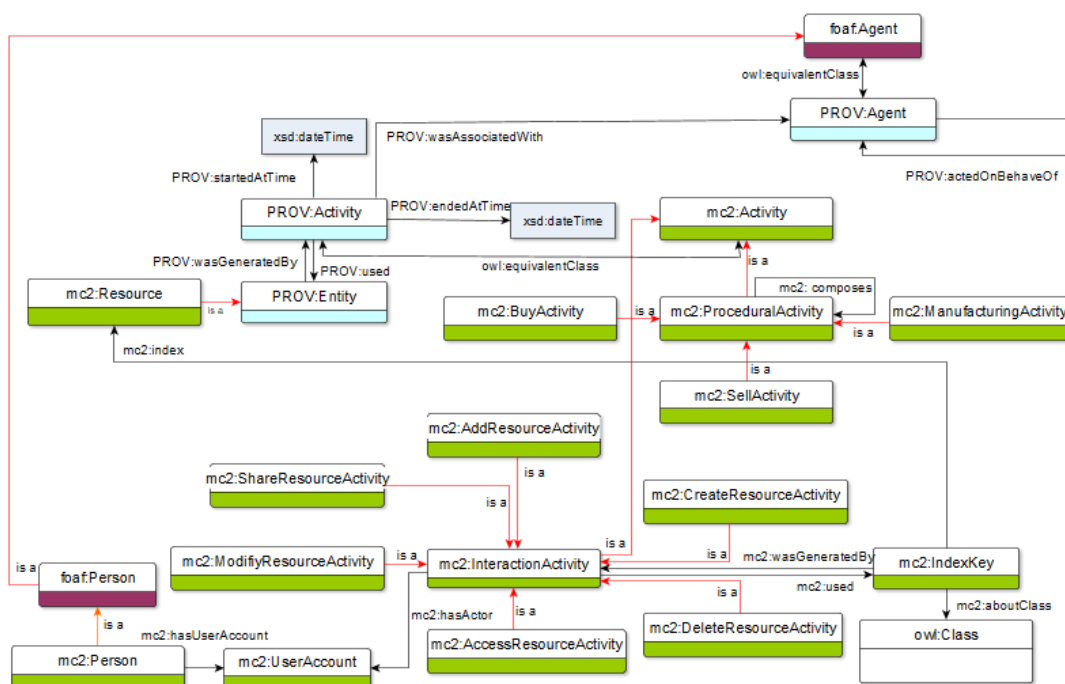


FIGURE 3.5: Activity module of MC2 (Atrash, 2015; Wang, 2016).

Besides, in the web platform *E-MEMORAE2.0* (see Figure 3.6), individuals can collaborate in organizations (Atrash, 2015; DeParis, 2013). An organization is considered as a collection of individuals belonging to different groups. Each group provides a sharing space for individuals, where they are allowed to add and share resources (e.g., notes, documents) in a common reference (represented as a semantic map in Figure 3.6). Specifically, all resources are indexed by the concepts in the semantic map. When a concept is selected, it becomes the focus concept. Then, the resources indexed by this focus concept are accessible to individuals, in different sharing spaces. Besides, individuals are also able to collaborate through informal communication and spontaneous production of knowledge, e.g., semantic wiki, chat, or forum (Atrash, 2015).

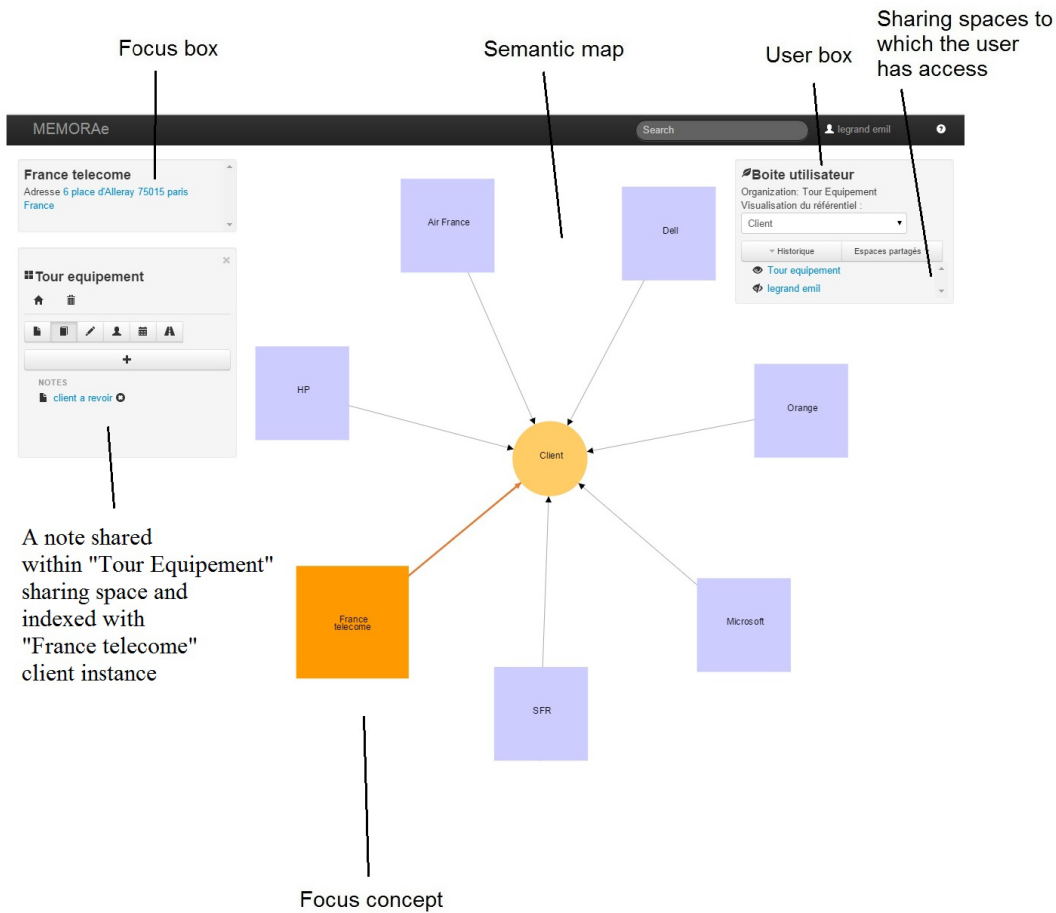


FIGURE 3.6: The main interface of E-MEMORAE2.0 (Atrash, 2015).

3.3 SOIS and MEMORAE SoIS

The core ontology *SOIS* is extended from MC2 (Saleh and Abel, 2016). It aims to aggregate and manage resources from different information systems (ISs) (Saleh, 2018). Thus, *SOIS* updated two main modules: **Resource** and **Activity**. All concepts in *SOIS* starts with the prex *sois* indicating its namespace.⁷

Resource

In a *SoIS*, resources are stocked in the information system where they were originally stored, whether a leader system (represented by *sois:LeaderSystem*) or a component system (represented by *sois:WebBasedApplication* and *sois:SandAloneSystem*). When a resource is contained in a component system, the leader system can provide users access to the resource through a reference key. In *SOIS*, each resource has a reference key to link itself and the

⁷The complete *SOIS* ontology is available at <https://gitlab.utc.fr/lisiying/ontologies-in-the-thesis-of-siyang-li.git>.

leader system (see Figure 3.7). Such a key can be an HTML tag, a Database Identifier, or a Hash tag.

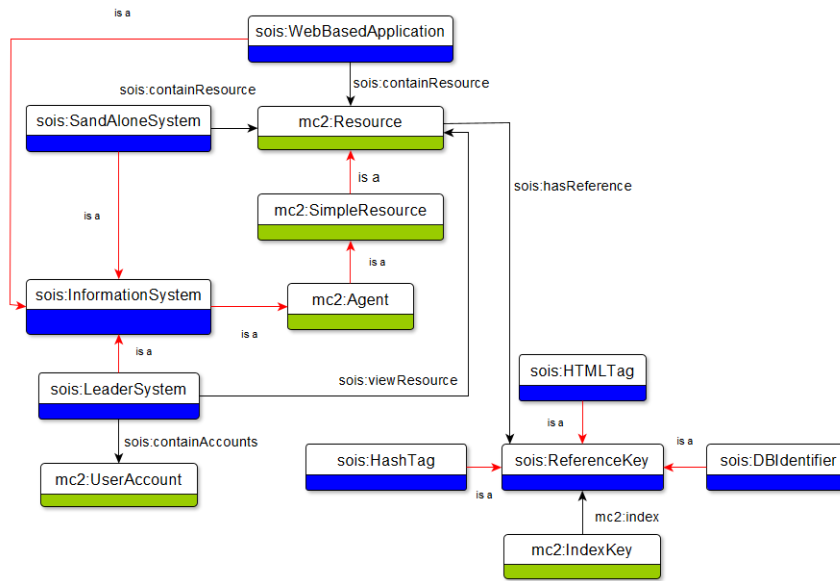


FIGURE 3.7: Resource module of SOIS (Saleh, 2018).

Activity

Users’ activities in the web platform is presented with the help of *ms:Activity*, which is derived directly from *mc2:Activity* (see Figure 3.8). SOIS considers five interaction activity types: accessing, deleting, voting, commenting and sharing. Besides, it also represents users’ accessing systems activity using *sois:NavigationActivity*. Similar to MC2, traces of activities defined in SOIS are also recorded and collected.

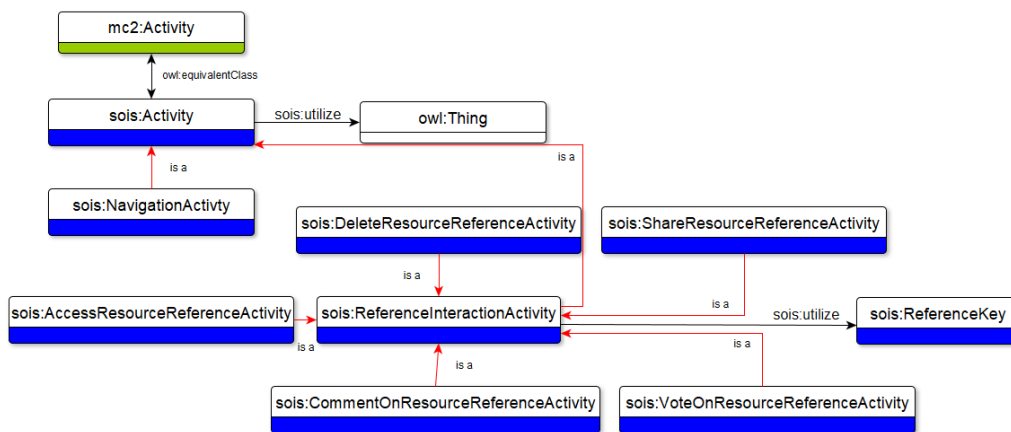


FIGURE 3.8: Activity module of SOIS (Saleh, 2018).

Besides, the web platform *MEMORAE SoIS* links to multiple autonomous external ISs for capitalizing the resources they can produce (Tiddlywiki, twitter, google contact, OneNote, etc.). These external ISs and *MEMORAE SoIS* together constitute a SoIS (see Figure 3.9),

where each resource is accessible in sharing spaces and indexed by the concepts in a semantic map.

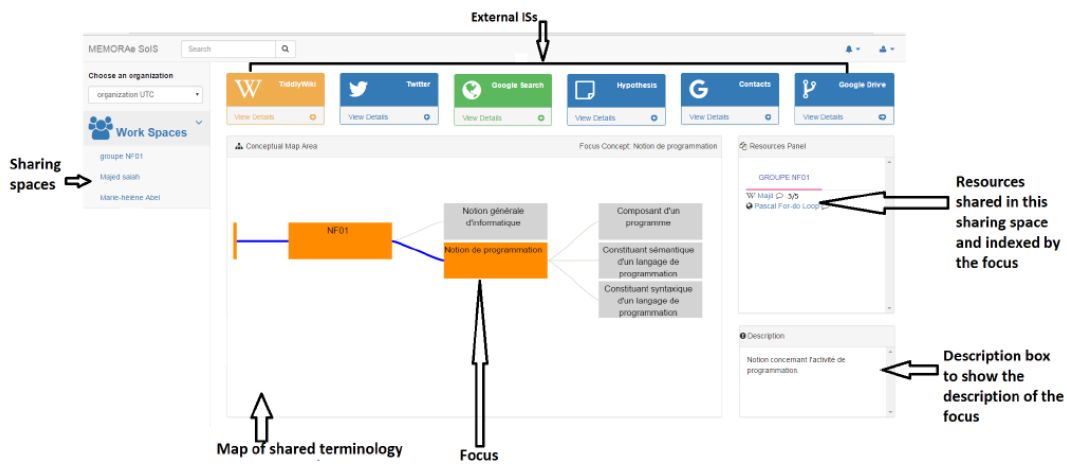


FIGURE 3.9: The main interface of MEMORAE SoIS (Saleh, 2018).

Particularly, the core ontology SOIS is used as a knowledge base of the leader system MEMORAE SoIS (see Figure 3.10).

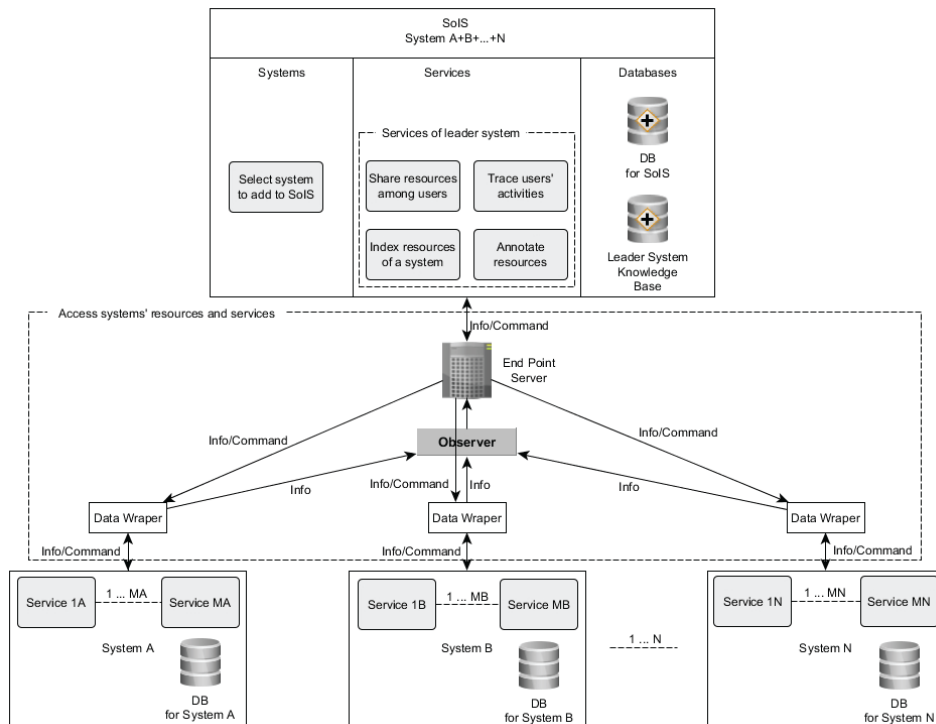


FIGURE 3.10: An architecture of collaborative SoIS (Saleh, 2018).

3.4 Discussion

Consisting of a core ontology and a web platform, the MEMORAE approach is suitable to construct and implement an ontology-based collaboration context model in web-based CWEs. Particularly, the reusability of ontology makes it possible to extend and composite the two existing ontologies (i.e., MC2 and SOIS) into an ontology-based collaboration context model, which can greatly simplify the construction process. The web platforms allows us to develop a prototype of web-based CWEs, where the ontology-based collaboration context model can be integrated to collect and process information within the context of users' collaborations.

Besides, both MC2 and SOIS are ontology-based models of collaboration. MC2 establishes the fundamental modules for collaborations between users in organizations. And SOIS deals with users' collaborations in a SoIS by accounting for the complexity and diversity of resource systems. This implies that MC2 and SOIS are worth reusing to build the ontology-based collaboration context model. Specifically, the main modules of MC2 and SOIS already cover several key impacting factors of collaboration (cf. Section 2.2). For instance, the two modules *resource* and *activity* correspond separately to the factors *resources* and *collaborators' actions*. Note that these factors are not completely represented and must be modified in the ontology-based collaboration context model.

However, MC2 and SOIS still have limitations respectively. Neither MC2 nor SOIS can support all users' collaborative activities in web-based CWE due to the incompleteness of their activity modules. MC2 does not consider vote or comment for resources, while SOIS does not support adding, creating, or modifying resources. Moreover, simply accumulating their activity modules together leads to redundant activities (access, share, and delete resources). This requires us to integrate and reorganize the two existing ontologies into the ontology-based collaboration context model. It must also be supplemented and extended according to the definition of the collaboration context, which will be presented in the rest of this thesis.

3.5 Chapter summary

In this chapter, we presented the MEMORAE approach and its compositions. After comparing its advantages and disadvantages, we decide to apply and continue the MEMORAE approach in this thesis, which can save our efforts in constructing and implementing an ontology-based collaboration context model in web-based CWEs.

In the next part, we will introduce our contributions in this thesis and explain how to generate relevant collaborator recommendations in web-based CWEs by constructing a Context-Aware Recommender System (CARS) that produces context-aware collaborator recommendations.

Part III

Contributions

Chapter 4

Collaboration context and its model

4.1 Introduction

To construct a Context-Aware Recommender System (CARS) that generates context-aware collaborator recommendations in a web-based Collaborative Working Environment (CWE), the first encountered problem is: what is the context of collaboration and how to model it? To address this, we need to clearly define context and collaboration context. It is also necessary to obtain an architecture of context models from which a well-designed collaboration context model can be established, which is also the core of CARS. Particularly, considering the advantages of the ontology-based modeling approach in terms of flexibility, extensibility, interpretability, reusability, and its support for interoperability between multiple information systems, we intend to construct an ontology-based collaboration context model (i.e., a collaboration context ontology) in web-based CWEs.

Therefore, the rest of this chapter is organized as follows. Section 4.2 presents our work on the definitions of context and collaboration context. Specifically, drawing on the definition of context, an architecture of context models is developed. Based on this architecture, we then explain how an ontology-based collaboration context model can be built and used in web-based CWEs (cf. Section 4.3). There is a summary at the end of this chapter.

4.2 Definitions

Based on the definitions of context (cf. Section 2.3.1), context contains the characteristic information that is pertinent to the circumstances of an event, statement, or object. When considering the event, statement or object as an entity, a user or application in the field of information technology can also be an entity (Dey, 2001). This indicates that the ranges of entities are varied in different fields, which we need to clarify in the definition of context. Besides, context is sensitive to time and its influence on the entity may shift over time. This should also be pointed out while defining context in this thesis. Therefore, context is identified as follows by supplementing Dey (2001)'s definition of context:

Context is any information that can be used to characterize the situation of an entity over a given period of time. An entity is a person, place, event or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves.

Any information considered as belonging to an entity's context is **contextual information**. It is also the value of **contextual factors**. For example, the value of factor *hour* is *17h*. Such factors can be grouped into **contextual dimensions** to describe the situation of an entity. For instance, dimension *time* may include four factors: *year*, *month*, *day*, and *hour*. These three interrelated concepts (contextual information, factor and dimension) can form an architecture of context models (see Figure 4.1).

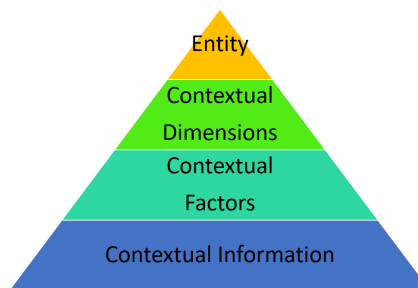


FIGURE 4.1: An architecture of context models.

The architecture (see Figure 4.1) describes an entity's context model. Particularly, this entity can be of different types. For example, for four types of entities: natural, human, artificial and group (see Table 2.7) (Zimmermann, Lorenz, and Oppermann, 2007), their contextual models respect this architecture. Thus, based on this architecture, any entity's context model can be developed by identifying the entity and defining its contextual dimensions, factors, and information. In this way, we are able to construct context models for all entities in web-based CWEs, regardless of the modeling approach.

Specifically, since the main purpose of web-based CWEs is to support users' collaborations, an indispensable entity in such environments is collaboration. Besides, defining the context of collaboration and building its model make it possible to gather more information about collaborations, thus helping users to understand their collaborations and to solve problems during collaborations. Therefore, in web-based CWEs, it is necessary to define the collaboration context and establish its model based on the architecture shown in Figure 4.1. Particularly, we determine to construct an ontology-based collaboration context model (i.e., a collaboration context ontology), given the advantages of the ontology-based modeling approach as discussed in Section 2.6.

To define the collaboration context, the definitions of collaboration should be investigated. Based on the literature review of collaboration (cf. Section 2.2.1), a collaboration

indicates two collaborators working together to accomplish their common goals through attempts. Therefore, collaboration can be defined as an event that involves at least two collaborators and consists of a set of actions carried out by the human actors acting on behalf of the corresponding collaborator, in order to achieve a common goal. Considering collaboration as an entity, the collaboration context can be defined precisely as follows, based on our above proposed definition of context:

The collaboration context is any information that can be used to characterize the situation of a collaboration over a given period of time. In web-based CWEs, collaboration is an event considered relevant to the interaction between users and/or applications, including the users and the applications themselves.

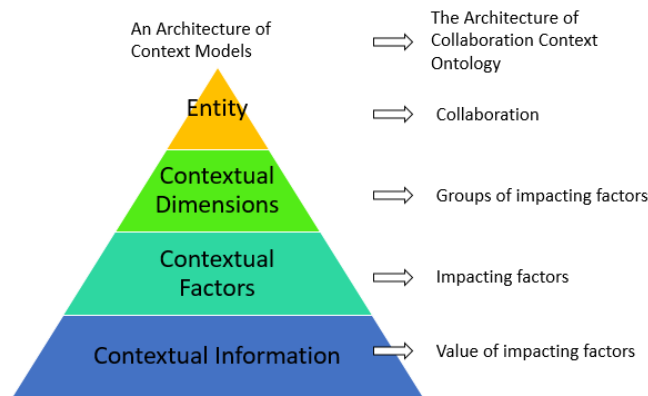


FIGURE 4.2: The architecture of the collaboration context ontology.

Next, to construct the ontology-based collaboration context model, we need to specify contextual dimensions, factors, and information of collaboration. While identifying the contextual factors, we need to explore the impacting factors of collaboration because of their influences on collaboration and its success (cf. Section 2.2). Notably, one of these impacting factors **context** affects other factors as well, requiring us to consider such impacting factors within the collaboration context. This signifies that these impacting factors can serve as contextual factors of collaboration. Thus, the contextual factors are represented by the impacting factors of collaboration. Groups of these impact factors are therefore equivalent to contextual dimensions, such as the groups in Table 2.1, 2.2 and 2.3. Besides, contextual information acts as the values of these factors. Hence, the ontology-based collaboration context model is composed of impacting factors, their values, and their groups (see Figure 4.2).

Specifically, we focus on eight groups of impacting factors: *Goal*, *Collaborator*, *Activity*, *Resource*, *Time*, *Location*, *Relation*, and *Satisfaction*. The first four groups are identified as a result of the analysis and discussion of shared impacting factors in Section 2.2.2. Next two groups *Time* and *Location* present two basic features of a collaboration: temporal and spatial. Since collaborations are connected, we also define an group *Relation* to encompass the factors that are relevant to these relationships between collaborations. Finally, the group

Satisfaction allows users to give their personal views about a collaboration. Each of these eight groups consists of several relevant impacting factors.

1. **Goal** includes multiple factors to describe desired products (outcomes) of a collaboration.
2. **Collaborator** is represented by factors about collaborator's abilities and demographic information.
3. **Activity** employ factors to specify types, actors, involved resources, time, and locations of activities during a collaboration.
4. **Resource** contains factors on resource types and identifications.
5. **Time** has factors to record start time and end time of a collaboration.
6. **Location** holds factors on identifications, types, and geographical information (e.g., longitude and latitude) of the places that are involved in a collaboration.
7. **Relation** comprises factors to stand for other related collaborations.
8. **Satisfaction** owns factors to represent collaborators' satisfaction degrees and comments about a collaboration.

With the identified contextual dimensions and factors, the ontology-based collaboration context model can be developed in web-based CWEs. Particularly, the process of building this model can be simplified by the existing ontologies of the MEMORAE approach (i.e., MC2 and SOIS) as discussed in Section 3.4. Thus, both MC2 and SOIS are extended and reused in the ontology-based collaboration context model.

4.3 Model: Collaboration context ontology

Based on MC2 and SOIS, an ontology-based collaboration context model (i.e. a collaboration context ontology) in web-based CWEs is constructed, known as MCC¹. It can:

1. Define contextual information, factors, and dimensions of collaboration
2. Describe users' collaborations and their contexts
3. Serve as a knowledge base
4. Process heterogeneous information from different sources

¹All concepts in MCC starts with the prex *mcc* indicating its namespace. The complete MCC ontology is available at <https://gitlab.utc.fr/lisiying/ontologies-in-the-thesis-of-siyang-li.git>.

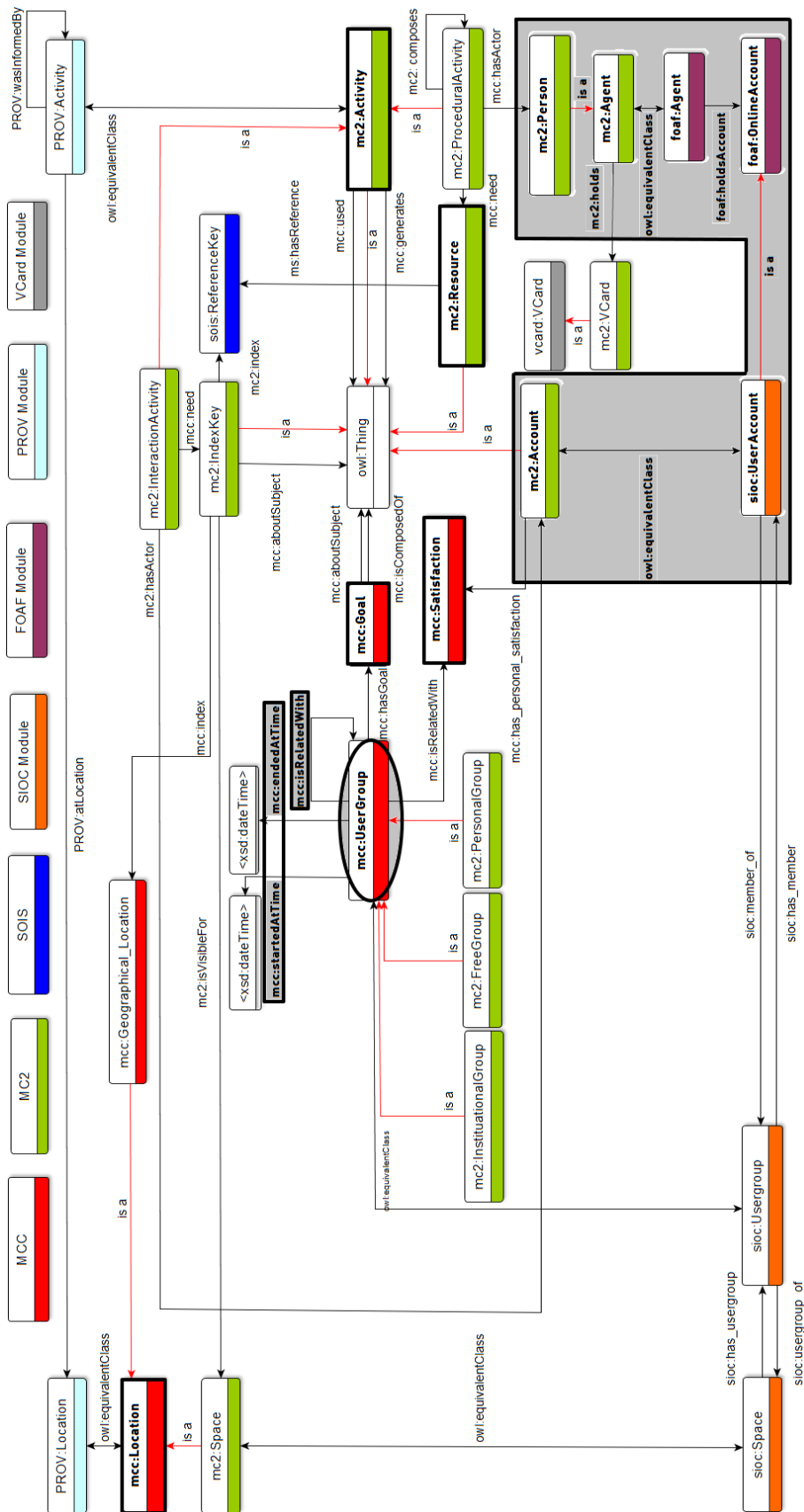


FIGURE 4.3: Eight contextual dimensions of collaboration in MCC (T-Box component).

This section explains the first two functionalities using the scenario presented in Section 1.2, where Emma collaborated with Lucie and Marinela in a context-aware application project. The rest two functionalities of MCC will be presented in Chapter 6.

To define collaborations, MCC applies a class of user group, `mcc:UserGroup` (see Figure 4.3). An instance of `mcc:UserGroup` indicates a collaboration between the members of the corresponding user group. In the scenario, Emma's collaboration in the context-aware application project can be represented by an instance `kb:Context_aware_application_project`². It provides a sharing space (i.e., `kb:Group_space_1`) for Emma, Lucie, and Marinela to interact with each other and to access available resources. Every `mcc:UserGroup` is held by a `mc2:Group`, equivalent to `foaf:Group`. This class represents a group in the real world. Members of a `mcc:Group`³ can collaborate multiple times (see Figure 4.4), while members of a `mcc:UserGroup` are limited to a single collaboration.

```

kb:Context_aware_application_project rdf:type mcc:UserGroup
kb:Context_aware_application_project rdf:type sioc:Usergroup
                                   (BY INFERENCE)

kb:Scientific_group rdf:type mcc:Group
kb:Scientific_group rdf:type foaf:Group (BY INFERENCE)
kb:Scientific_group mcc:holdsUserGroup
                   kb:Context_aware_application_project

kb:Emma rdf:type mc2:Person
kb:Emma rdf:type mc2:Agent (BY INFERENCE)
kb:Emma rdf:type foaf:Agent (BY INFERENCE)
kb:Scientific_group foaf:member kb:Emma

kb:Lucie rdf:type mc2:Person
kb:Scientific_group foaf:member kb:Lucie

kb:Marinela rdf:type mc2:Person
kb:Scientific_group foaf:member kb:Marinela

kb:Group_space_1 rdf:type mc2:Space
kb:Group_space_1 rdf:type sioc:Space (BY INFERENCE)
kb:Group_space_1 sioc:has_usergroup kb:Scientific_project
kb:Context_aware_application_project sioc:usergroup_of
                                     kb:Group_space_1

```

Surrounding `mcc:UserGroup`, the eight contextual dimensions of collaboration (i.e., Goal,

²All instances of the scenario starts with the prex `kb` indicating its namespace.

³In MCC, a `mcc:Group` can create multiple instances of `mcc:UserGroup` to stand for different collaborations.

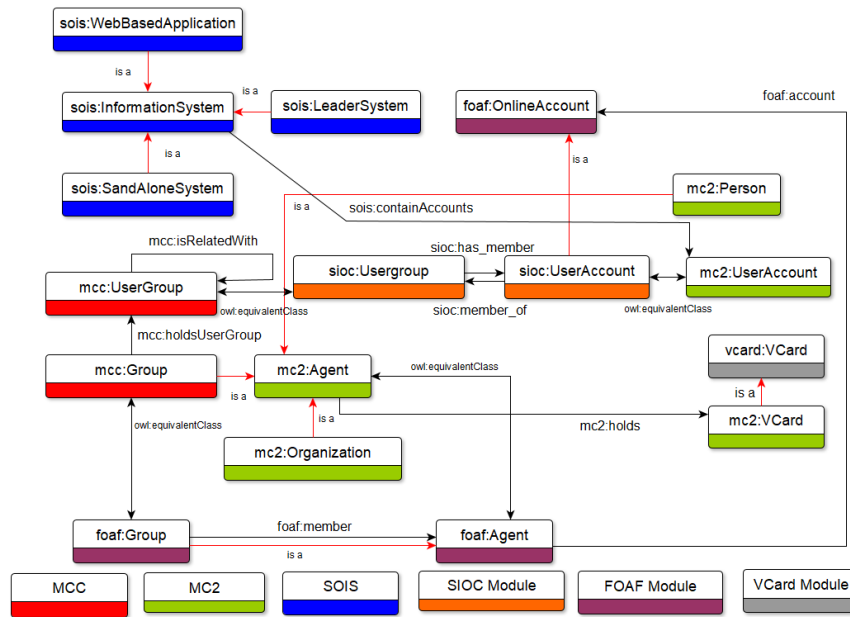


FIGURE 4.4: The dimension *Collaborator* in MCC (T-Box component).

Collaborator, Activity, Resource, Time, Location, Relation, and Satisfaction) are represented by different classes and/or their interrelationships, as specified with rectangles in Figure 4.3.

In MCC, a `mcc:UserGroup` includes a set of collaborators (at least two) that have a common goal to achieve and hold their own user accounts. These collaborators are human actors acting on behalf of themselves, groups or even organizations, which is detailed in Figure 4.4. In the scenario, the instance of `mcc:UserGroup` (i.e., `kb:Context_aware_application_project`) contains user accounts of Emma, Lucie, and Marinela, rather than themselves. These users accounts are managed in a web-based CWE (represented by `sois:LeaderSystem`). Through their accounts, Emma, Lucie, and Marinela can access to different collaborative tools (represented by `sois:StandAloneSystem` and `sois:WebBasedApplication`) that are integrated into the web-based CWE.

```

kb:Emma_account rdf:type mc2:Account
kb:Emma_account rdf:type sioc:UserAccount (BY INFERENCE)
kb:Emma_account rdf:type foaf:OnlineAccount (BY INFERENCE)
kb:Emma foaf:holdsAccount kb:Emma_account
kb:Emma_account sioc:member_of kb:Scientific_group
kb:Scientific_group sioc:has_member kb:Emma_account

kb:Lucie_account rdf:type mc2:Account
kb:Lucie foaf:holdsAccount kb:Lucie_account
kb:Lucie_account sioc:member_of kb:Scientific_group
kb:Scientific_group sioc:has_member kb:Lucie_account

kb:Marinela_account rdf:type mc2:Account
kb:Marinela foaf:holdsAccount kb:Marinela_account

```

```

kb:Marinela_account sioc:member_of kb:Scientific_group
kb:Scientific_group sioc:has_member kb:Marinela_account

```

Besides, a `mcc:UserGroup` has a common goal among its members. Such a goal is described by multiple concepts, expressed as `owl:Thing` in MCC. In the scenario, the goal of the context-aware application project is to build a context-aware application and to publish a corresponding paper (see Figure 4.5).

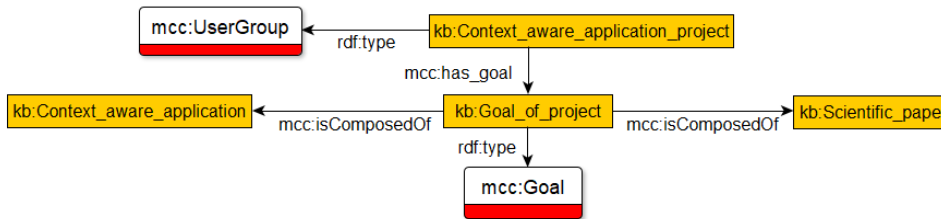


FIGURE 4.5: The goal of the context-aware application project in the scenario (A-Box component).

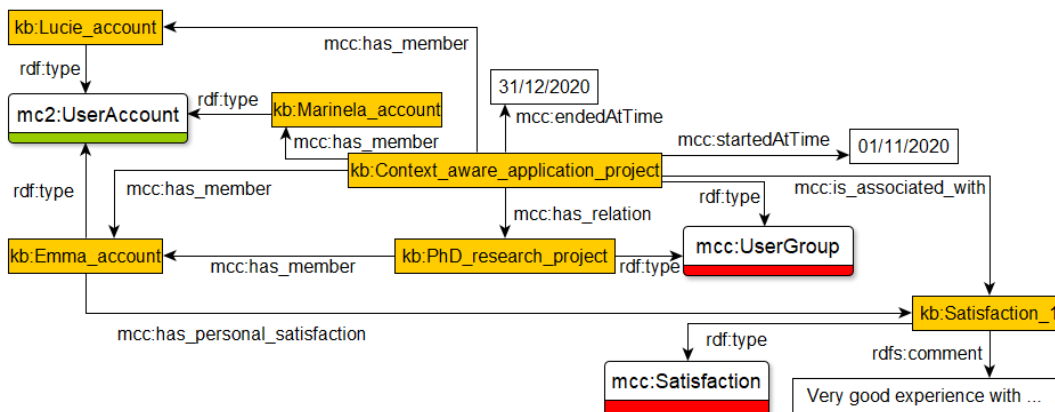


FIGURE 4.6: *Time, Relation and Satisfaction* representations in the scenario (A-Box component).

For the other three dimensions (Time, Relation and Satisfaction), they are all directly related to `mcc:UserGroup`. With them, MCC is able to register collaborator's feedback and analyze relationship between different collaborations. In the scenario, the context-aware application project is a part of Emma's PhD research with her professors. Thus, the two instances of `mcc:UserGroup` are connected with each other (see Figure 4.6). Besides, Figure 4.6 also demonstrates the common collaborators, their satisfactions, start time, and end time of the two instances.

MCC also applies MC2, SOIS, FOAF, SIOC, and VCard to model the contextual dimension - Collaborator (see Figure 4.4). Especially, MC2 and VCard aim to construct profiles for collaborator, including their demographic information and abilities (see Figure 4.7). In

the scenario, to arrange tasks between Emma, Lucie, and Marinela, their profiles can be consulted. Figure 4.8 illustrates Marinela’s profile, indicating that she can develop interfaces of the context-aware application.

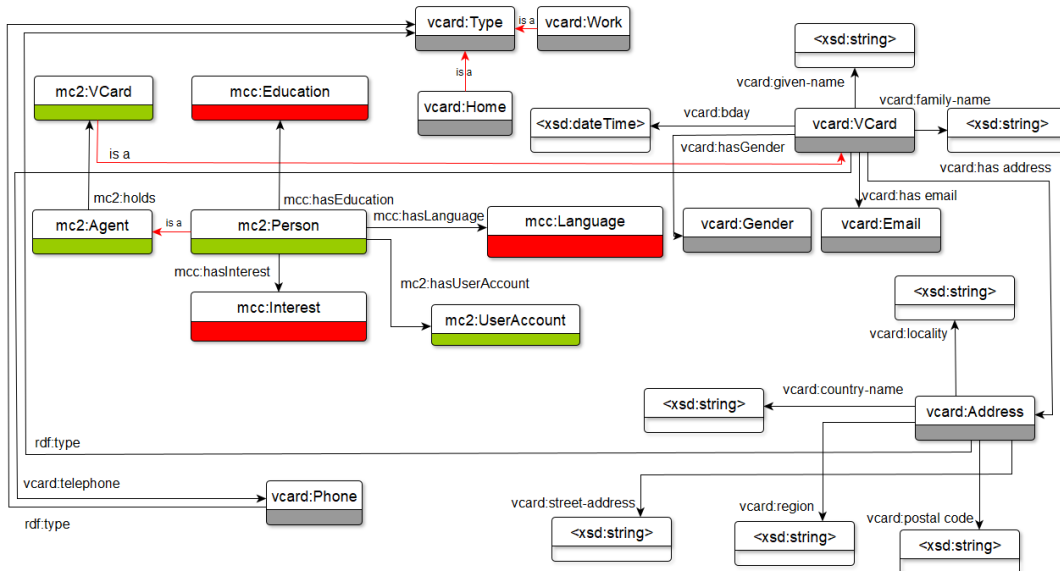


FIGURE 4.7: Collaborator profile in MCC (T-Box component).

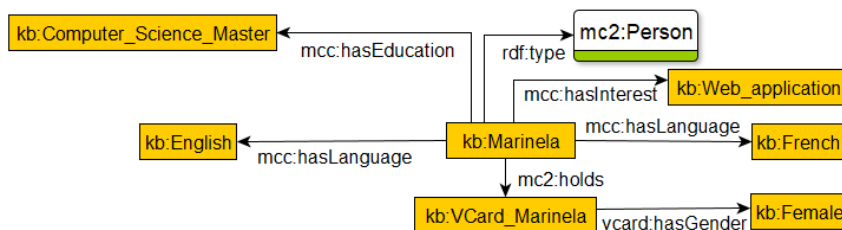
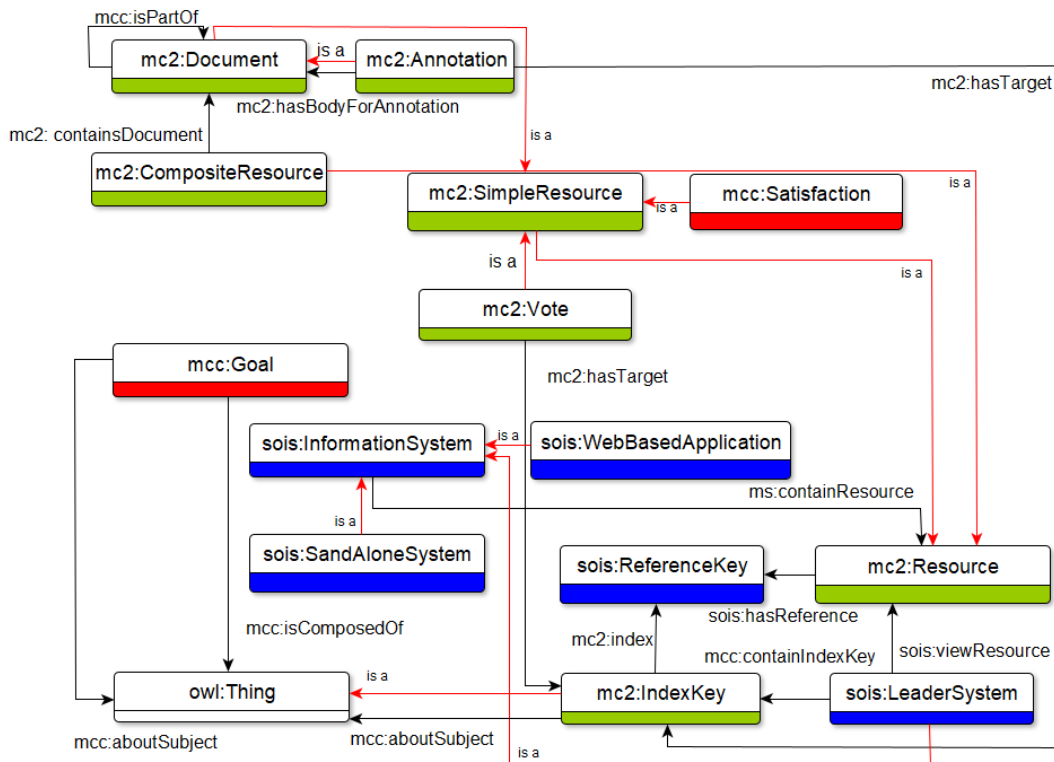


FIGURE 4.8: Marinela’s profile in the scenario (A-Box component).

As for *Resource*, MCC divides them into two categories based on their complexity levels: simple and composite (see Figure 4.9). Particularly, resources can be contained either in a web-based CWE, or in an integrated collaborative tool. To associate resources in integrated systems with the web-based CWE, `sois:ReferenceKey` is applied and included in the CWE. Each `sois:ReferenceKey` and resource stored in the web-based CWE has a `mc2:IndexKey` that is visible to certain user groups. Thus, through `mc2:IndexKey` and `sois:ReferenceKey`, all resources are accessible and visible in user groups of the web-based CWE, regardless of where they are deposited. When user interacting with different resources in the web-based CWE, it is their reference keys and index keys that are used and modified. Moreover, each resource is indexed with a concept of collaboration goals (represented by `owl:Thing`). This signifies that resource in the web-based CWE are tied to the collaboration goals through MCC.

FIGURE 4.9: The dimension *Resource* in MCC (T-Box component).

Then, MCC integrates the two activities modules (in MC2 and SOIS) together and classifies activities into three categories (see Figure 4.10). Particularly, navigational activities (represented by `sois:NavigationActivity`) involve browsing information systems and navigation concepts of collaboration goals. Users' interaction activities on resource (represented by `mc2:InteractionActivity`) are reorganized and updated (e.g. annotating and discussing resources). Besides, MCC makes it possible to record users' activities outside the web-based CWE (represented by `mc2:ProceduralActivity`). In the scenario, when Emma wrote the scientific paper about their context-aware application (see Figure 4.11), she cited a previously published conference paper written by herself. Her activity of writing the conference paper did not take place in the web-based CWE, but can be imported into it as an instance of `mc2:ProceduralActivity`. In MCC, any activity can utilize and/or generate anything, such as resources. For example, Emma utilizes the conference paper and generates the scientific paper in her activity of writing the scientific paper (see Figure 4.11).

Furthermore, an activity is associated with a location, represented by `mcc:Location` (see Figure 4.12). Specifically for geographical locations, their longitude and latitude are contained in MCC. Within information on locations, MCC can track what resources are used in which activity at which location, such as the resources and locations involved in Emma's activity of writing the scientific paper (see Figure 4.11).

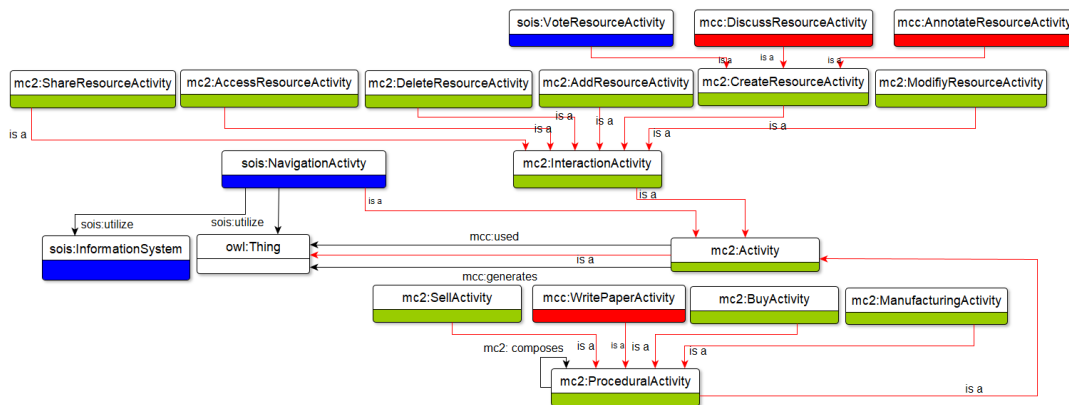


FIGURE 4.10: The dimension Activity in MCC (T-Box component).

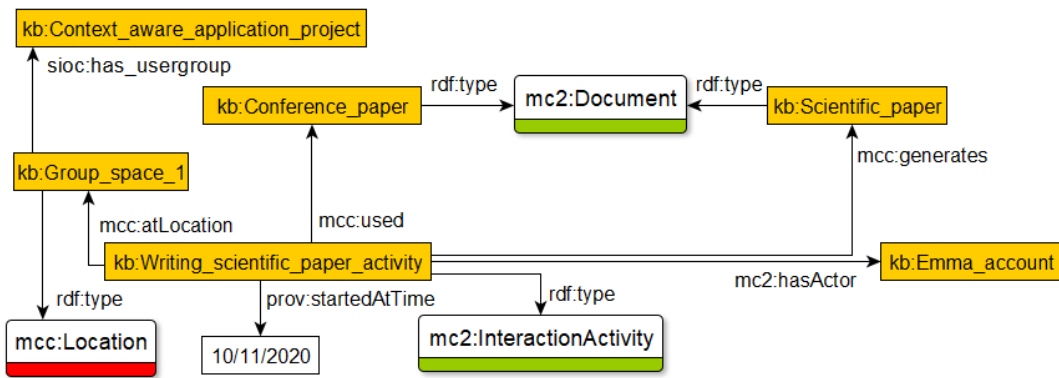


FIGURE 4.11: Emma's creating resources activity (A-Box component).

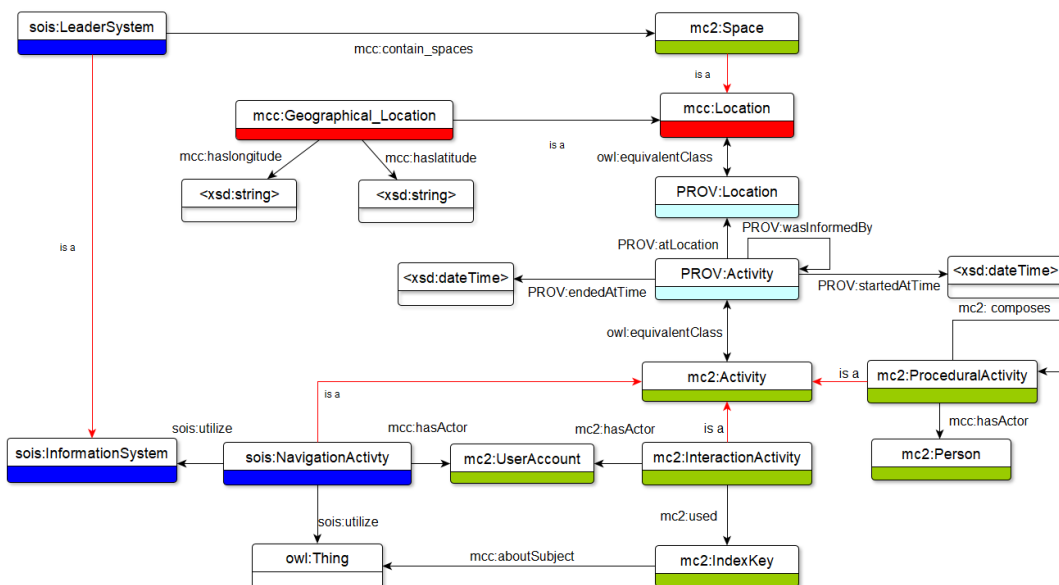


FIGURE 4.12: Relation between dimensions Location and Activity in MCC (T-Box component).

Using MCC, users' collaborations and their contexts in web-based CWEs can be presented around `mcc:UserGroup`. Particularly, users can examine the success of their collaborations depending on the contextual dimension Goal. Meanwhile, the other seven dimensions can serve to access the efficiency of their collaborations, such as collaborators' contributions, productivity of the performed activities, and utilities of the involved resources.

4.4 Chapter summary

In this chapter, we focused on defining and modeling the collaboration context in web-based CWEs, enabling us to answer the first question listed in Section 1.1.3: what is collaboration context and how to model it.

Particularly, we first proposed a definition of context and an architecture of context models based on three interrelated concepts: contextual information, contextual factor, and contextual dimension. This then allows us to define the collaboration context and, based on the architecture, to construct an ontology-based collaboration context model (i.e., a collaboration context ontology), MCC. Finally, we explained how MCC can be applied to represent users' collaborations and their contexts in web-based CWEs.

Within MCC, the next chapter will present how to incorporate the collaboration context into the recommendation generation process and how to generate context-aware collaborator recommendations.

Chapter 5

Context-aware collaborator recommendations

5.1 Introduction

With the definition and the ontology-based model of collaboration context, our next step is to work out how to they can be processed and employed in algorithms for generating context-aware collaborator recommendations? To achieve this, we need to integrate the collaboration context contained in the ontology-based model into the collaborator recommendation generation processes. For this, three methods are available: Pre-Filtering (PreF), Post-Filtering (PoF), and Contextual Modeling (CM) (Adomavicius and Tuzhilin, 2011). Among these methods, CM has limitations in computing multi-dimensional data when the volume and complexity of the data are large as discussed in Section 2.5. Therefore, we focus on applying PreF and PoF in this thesis to generate context-aware collaborator recommendations for users in web-based CWEs. Based on the literature review (cf. Section 2.5.2), the main phases of the two methods are shown in Figure 5.1.

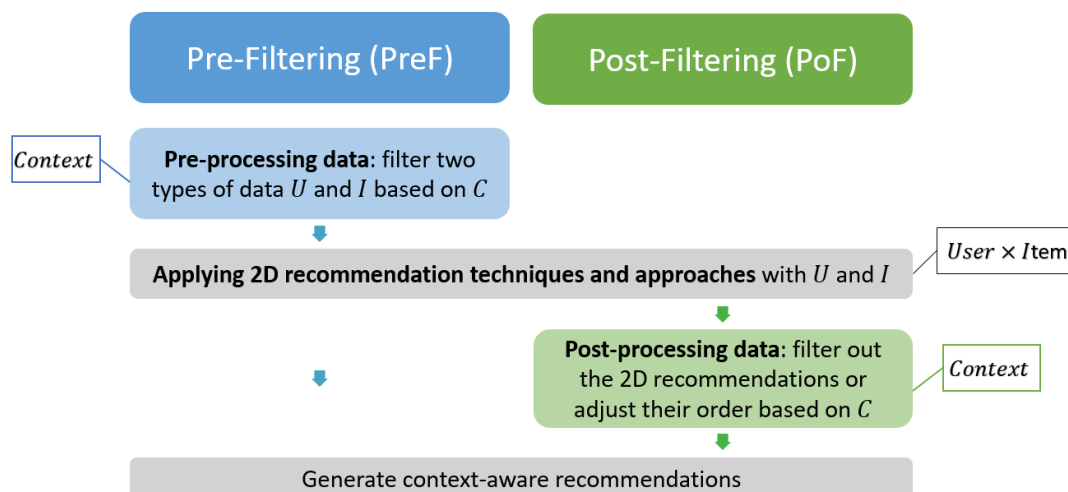


FIGURE 5.1: Main phases of PreF and PoF.

Particularly, three common phases of PreF and PoF can be resumed, regardless of their order:

- Employ $User(U)$ and $Item(I)$ to generate 2D recommendations
- Process $Context(C)$ (pre-processing in PreF and post-processing in PoF)
- Produce context-aware recommendations

Following these common phases of PreF and PoF, three major pieces of work on context-aware collaborator recommendations are needed: 1) generate 2D collaborator recommendations using $User(U)$ and $Item(I)$; 2) process the collaboration context (i.e., $Context(C)$); 3) produce context-aware collaborator recommendations.

Therefore, the rest of this chapter is organized as follows. Section 5.2 first introduces existing studies of a 2D recommendation algorithm and several ontology-based semantic similarities. Section 5.3 presents how to generate 2D collaborator recommendations. Then, we develop and utilize an ontology-based semantic similarity to process the collaboration context (cf. Section 5.4). Next, Section 5.5 shows two algorithms and explains how they can be used to produce context-aware collaborator recommendations. Finally comes the summary of this chapter.

5.2 Preliminary

5.2.1 Probabilistic matrix factorization

Probabilistic Matrix Factorization (PMF) (Mnih and Salakhutdinov, 2008) is a 2D recommendation algorithm, belonging to model-based CF approach (discussed in Section 2.5.1.2). It was proposed and developed from Matrix Factorization (MF) (Koren, Bell, and Volinsky, 2009; Baltrunas, Ludwig, and Ricci, 2011).

When generating 2D recommendations, MF (Koren, Bell, and Volinsky, 2009; Baltrunas, Ludwig, and Ricci, 2011) needs a rating matrix R of $m \times n$ that contains users' ratings¹ for items (m users and n items). Given such a rating matrix R , MF factorizes it into two matrices U of $m \times d$ and I of $n \times d$. Usually, $d (< \min(m, n))$ refers to the number of latent factors that characterize users and items. For example, given a rating matrix R of 4×3 , d can equal to 1 or 2. Then, a user is represented by a j^{th} row vector in user matrix $U(U_j \in U)$ and an item by a k^{th} column vector in item matrix $I(I_k \in I)$. Finally, MF predicts the unknown rating \widehat{R}_{jk} that user j might give for item k following Equation 5.1:

¹Usually, these ratings are non-negative. This indicates that $R_{jk} \geq 0$. If $R_{jk} = 0$, then the user j has not yet given his/her rating to the item k .

$$\widehat{R}_{jk} = U_j I_k + \bar{r}_k + b_j + b_k \quad (5.1)$$

where \bar{r}_k is the average rating of item k in the rating matrix R ; b_j and b_k indicates the observed deviations of user j and item k . The predicted rating is broken into its four components: global average (\bar{r}_k), item bias (b_k), user bias (b_j), and user-item interaction ($U_j I_k$).

Based on MF, PMF adopts a probabilistic linear model with Gaussian observation noise (Mnih and Salakhutdinov, 2008) on the unknown ratings $R_{jk}^* (= U_j I_k)$. Its key idea is to treat rating prediction as a generative process and define a conditional probability over the known ratings (represented by positive ratings in R (Zhu, Shen, and Zhou, 2019)). In PMF, $R_{jk} - R_{jk}^*$ is normally distributed with mean 0 and variance σ^2 . Then the conditional probability of R_{jk}^* can be defined (see Equation 5.2) (Mnih and Salakhutdinov, 2008).

$$p(R|U, I, \sigma^2) = \prod_{j=1}^m \prod_{k=1}^n [\mathcal{N}(R_{jk}|U_j I_k, \sigma^2)]^{V_{jk}} \quad (5.2)$$

where $R_{jk} \in [1, Q]$, Q is the maximum real-valued rating; V_{jk} is the indicator function that is equal to 1 if user j rated item k and equal to 0 otherwise; the mean of $[\mathcal{N}(R_{jk}|U_j I_k, \sigma^2)]^{V_{jk}}$ is $U_j I_k V_{jk}$ and variance $\sigma^2 V_{jk}$.

Besides, Mnih and Salakhutdinov (2008) also placed zero-mean spherical Gaussian priors on user and item feature vectors. This implies that user and item feature vectors are also normally distributed. Their conditional probabilities therefore are:

$$\begin{aligned} p(U|\sigma_U^2) &= \prod_{j=1}^m \mathcal{N}(U_j|0, \sigma_U^2 V) \\ p(V|\sigma_I^2) &= \prod_{k=1}^n \mathcal{N}(I_k|0, \sigma_I^2 V) \end{aligned} \quad (5.3)$$

where the mean of $\mathcal{N}(U_j|0, \sigma_U^2 V)$ is 0 and the variance $\sigma_U^2 V$; the mean of $\mathcal{N}(I_k|0, \sigma_I^2 V)$ is 0 and the variance $\sigma_I^2 V$.

Then, to get the values of user feature vectors, item feature vectors and unknown ratings, maximum likelihood estimation and maximum a posteriori estimation are applied. The likelihood function (see Equation 5.4) is about two parameters U and I , where others are hyper-parameters. The goal is to minimize Equation 5.4 (Mnih and Salakhutdinov, 2008), which is equivalent to maximizing the log-posterior of Equation 5.2 over users and items with Equation 5.3.

$$\mathcal{L}(U, I) = \frac{1}{2} \prod_{j=1}^m \prod_{k=1}^n V_{jk} (R_{jk} - U_j I_k)^2 + \frac{\lambda_U}{2} \prod_{j=1}^m \|U_j\|_{Fro}^2 + \frac{\lambda_I}{2} \prod_{k=1}^n \|I_k\|_{Fro}^2 \quad (5.4)$$

where $\lambda_U = \sigma^2 / \sigma_U^2$, $\lambda_I = \sigma^2 / \sigma_I^2$, and $\|\cdot\|_{Fro}^2$ denotes the Frobenius norm.

A local minimum of the function in Equation 5.4 can be found by performing gradient descent algorithm in Appendix A (Mnih and Salakhutdinov, 2008).

Particularly, the predicted ratings in PMF sometimes can exceed the range of valid rating values ($[1, Q]$) (Mnih and Salakhutdinov, 2008). Thus, to bound the range of predicted ratings, Mnih and Salakhutdinov (2008) proposed to apply the logistic function $\mathcal{G}(U_j I_k) = \frac{1}{1 + \exp(-(U_j I_k))}$, instead of directly using $U_j I_k$ in Equation 5.2 and 5.4. They also mapped the known ratings R_{jk} to the interval $[0, 1]$ using the function $\mathcal{T}(R_{jk}) = (R_{jk} - 1)/(Q - 1)$, so that the range of known ratings can match the range of predicted ratings in PMF. Accordingly, the conditional probability of ratings transforms from Equation 5.2 to Equation 5.5:

$$p(R|U, I, \sigma^2) = \prod_{j=1}^m \prod_{k=1}^n [\mathcal{N}(R_{jk} | \mathcal{G}(U_j I_k), \sigma^2)]^{V_{jk}} \quad (5.5)$$

5.2.2 Ontology-based semantic similarity

Semantic similarity is a measurement that aims to compute the likeness/relatedness between classes (instances) and their relationships in knowledge bases (Sánchez et al., 2012; Batet, Sánchez, and Valls, 2011), helping to integrate knowledge into the data mining process (Zhang et al., 2008). Thanks to the possibilities that ontologies can serve as knowledge bases, ontology-based semantic similarities have recently been exploited. Such similarities can be classified into three types (Sánchez et al., 2012; Batet, Sánchez, and Valls, 2011; Zhang et al., 2008; Ovaska, Laakso, and Hautaniemi, 2008): path-based, feature-based, and information content-based semantic similarity.

While measuring path-based semantic similarities, an ontology is seen as a directed graph where classes (instances) are interrelated mainly by means of various relationships (e.g., is-a) (Sánchez et al., 2012). In such a graph, semantic similarities are usually calculated based on the shortest path between two classes (instances) (Zhang et al., 2008; Rada et al., 1989; Batet, Sánchez, and Valls, 2011). The longer the shortest path, the more semantically different the two classes (instances) are (Sánchez et al., 2012). This implies that path-based semantic similarities don't necessitate the detailed information of each class (instance), which is an advantage of such similarities. However, their main problem is that they have a strong dependence on the degree of completeness, homogeneity and coverage of the relationships in the ontology (Cimiano, 2006).

Besides, in the calculation of feature-based semantic similarities, classes (instances) in ontologies are described by a set of ontological features (Zhang et al., 2008; Sánchez et al., 2012). The more common features and the less non-common features two classes (instances) have, the more similar they are (Varelas et al., 2005). To compare features, several coefficients on the sets are applicable to feature-based semantic similarities, such as Jaccard index

$Jaccard(O^x, O^y) = \frac{|O^x \cap O^y|}{|O^x| + |O^y| - |O^x \cap O^y|}$ ² (Jaccard, 1901), Dice coefficient $Dice(O^x, O^y) = \frac{2|O^x \cap O^y|}{|O^x| + |O^y|}$ ³ (Dice, 1945), and Tversky index $Tversky(O^x, O^y) = \frac{|O^x \cap O^y|}{|O^x \cap O^y| + \alpha|O^x - O^y| + \beta|O^y - O^x|}$ ⁴ (Tversky, 1977). This indicates that unlike path-based similarities, feature-based similarities require detailed information about the features of each class (instance). Their one advantage is that they can be employed in cross ontologies (i.e. when the two classes (instances) belong to different ontologies), but path-based similarities cannot (Petrakis et al., 2006).

As for information content-based semantic similarities, they measure the amount of information provided by a common ancestor of two classes (instances) in an ontology (Batet, Sánchez, and Valls, 2011; Zhang et al., 2008; Sánchez et al., 2012). Particularly for a class (instance) x , such similarities utilize $IC(x) (= -\log p(x))$ ⁵ to identify the amount of its provided information. Using $IC(x)$, infrequent classes (instances) are more informative than the frequent ones (Sánchez et al., 2012). However, such semantic similarities request recursive computation of all classes' and instances' appearances in an ontology. If any class (instance) or their relations change, recalculations are mandatory. This also implies that information content-based semantic similarities are inapplicable in cross ontologies, lowering the scalability of such similarities (Batet, Sánchez, and Valls, 2011).

Beyond the mentioned similarities, there are other semantic similarities as well. For example, Batet, Sánchez, and Valls (2011) proposed a semantic similarity that combines feature-based and information content-based similarities. Carrer-Neto et al. (2012) employed Equation 5.6 to calculate the semantic similarity based on the features of classes (instances) and the information provided by their relationships.

$$Similarity_{semantic}(x, y) = \sum_{g=1}^{|P^x \cap P^y|} \frac{|O_{p_g}^x \cap O_{p_g}^y|}{\max(|O_{p_g}^x|, |O_{p_g}^y|)} \times Weight(p_g) \quad (5.6)$$

where $|P^x \cap P^y|$ indicates the number of relationships that both x and y have; $|O_{p_g}^x|$ represents the number of classes (instances) associated to x through the relationship p_g ; $|O_{p_g}^x \cap O_{p_g}^y|$ represents the number of common classes (instances) associated to x and y through the relationship p_g ; $Weight(p_g)$ expresses the importance of the relationship p_g .

5.3 2D collaborator recommendations

To generate 2D collaborator recommendations, a 2D recommendation approach should be selected. Based on the literature review of 2D recommendation approaches (cf. Section

²Here, x and y are two classes (instances) in ontologies; O^x and O^y refer to their sets of ontological features; $|O^x \cap O^y|$ the number of common features in the sets O^x and O^y ; $|O^x|$ denotes the number of features in the set O^x . The range of $Jaccard(O^x, O^y)$ is $[0, 1]$.

³The range of $Dice(O^x, O^y)$ is $[0, 1]$.

⁴Here, $O^x - O^y$ denotes the relative complement of O^y in O^x . $\alpha, \beta \geq 0$ are parameters of the Tversky index. Setting $\alpha = \beta = 1$ produces Jaccard index; setting $\alpha = \beta = 0.5$ produces Dice coefficient.

⁵Here, $p(x)$ is the probability of x 's appearance in the ontology.

2.5.1), four main approaches can be applied: Content-Based filtering (CB), Collaborative Filtering (CF), Knowledge-Based (KB), and Hybrid.

First, due to the data entry conflict, CB approach is not appropriate. Specifically, CB needs information to construct collaborator (i.e., item) descriptions and user profiles to generate 2D collaborator recommendations (Lops, De Gemmis, and Semeraro, 2011; Adomavicius et al., 2005). However, such information belongs to the collaboration context ($Context(C)$ in Figure 5.1), instead of users and collaborators ($User(U)$ and $Item(I)$ in Figure 5.1). Thus, CB is inappropriate for generating 2D collaborator recommendations in web-based CWEs.

Then, KB approach is also unsuited because of the data entry conflict and its extra requirements. To generate 2D collaborator recommendations, KB additionally necessitates knowledge bases. The information contained in such knowledge bases is part of collaboration context ($Context(C)$ in Figure 5.1), such as collaborator (i.e., item) descriptions (Bridge et al., 2005; Felfernig et al., 2011), resulting in data entry conflict. Besides, the construction of knowledge bases makes KB non-universal and high cost, having no interest for us.

Moreover, Hybrid approach can't be implemented, owing to its nature. Such approaches must combine two or more other native approaches (CB, CF, and KB) (Adomavicius and Tuzhilin, 2005; Burke, 2002). However, recognizing the shortcomings of the two native approaches (CB and KB), only one native approach is applicable. It is therefore impossible to realize Hybrid approaches for generating 2D collaborator recommendations.

Finally, in CF approach, model-based CF generates 2D recommendations just with users' ratings for items (Breese, Heckerman, and Kadie, 1998). This suggests that only $User(U)$ and $Item(I)$ are used in this approach (see Figure 5.1), there is no data entry conflict in applying model-based CF. Plus, we're in a web-based CWE where users collaborate with each other in corresponding user groups. In such environments, model-based CF approaches have already been used to generate recommendations (Wang et al., 2012). Thus, we decide to employ model-based CF for generating 2D collaborator recommendations for users in web-based CWEs. Specifically, model-based CF involves many techniques: Bayesian networks (Breese, Heckerman, and Kadie, 1998), Markov decision process (Shani, Heckerman, and Brafman, 2005), MF (Koren, Bell, and Volinsky, 2009), Latent Dirichlet Allocation (LDA) (Blei, Ng, and Jordan, 2003), Singular Value Decomposition (SVD) (Billsus and Pazzani, 1998). Among all these techniques, we turn toward PMF (Mnih and Salakhutdinov, 2008), developed from MF. This is mainly for three reasons.

1. As techniques in model-based CF approach, both PMF and MF can be directly integrated in PreF and PoF. Besides, MF is even capable of generating recommendations with several types of context, such as Collective MF (CMF) (Singh and Gordon, 2008) and context-adaptive MF (AdaMF) (Man et al., 2015).
2. MF is a widely used technique in generating 2D recommendations and context-aware recommendations (Raza and Ding, 2019). It has excellent performances in the Netflix

Prize⁶ competition (Adomavicius et al., 2005) and provides a foundation for other algorithms, such as Tensor Factorization (TF) (Karatzoglou et al., 2010) and Sparse Linear Model (SLIM) (Ning and Karypis, 2011).

3. As an evolved version of MF, PMF was proposed and proved to be effective and accurate (Mnih and Salakhutdinov, 2008). PMF considers the rating as a random variable and applies Gaussian distribution as a conditional probability over the known ratings, which performed well on the large, sparse, and very imbalanced Netflix dataset (Mnih and Salakhutdinov, 2008). This indicates that PMF becomes a new milestone in the development of MF since it solve sparsity problem and makes MF a technique that only has cold-start problem. Based on PMF, many enhancements and derived applications have been studied, such as Bayesian PMF (BPMF) (Salakhutdinov and Mnih, 2008) and Generalized PMF (Shan and Banerjee, 2010). Ma et al. (2008) also applied PMF to solve social recommendation problems.

Therefore, PMF is chosen to generate 2D collaborator recommendations for users in web-based CWEs. Correspondingly, its rating function $R_{collaborator}$ is:

$$R_{collaborator} : User \times Collaborator \rightarrow Rating \quad (5.7)$$

where for a user u in a user group c , a collaborator is another user $i (\neq u)$ that is not in the user group c .

Based on the principles of PMF (cf. Section 5.2.1), we need to construct a rating matrix R with users' known ratings to predict their unknown ratings. Particularly in the 2D collaborator recommendation problem, collaborators are also users. Thus, we have m users and m collaborators in the rating matrix R . Consequently, the dimension of R is $m \times m$, while the dimensions of the user matrix U and collaborator matrix I are both $m \times d$.⁷

In the rating matrix R , R_{jk} (see Equation 5.8)⁸ is a user j 's rating for collaborator k , which indicates user j 's collaboration frequency with collaborator k .

$$R_{jk} = \begin{cases} \frac{e_{jk}-1}{E_{max}-1}, & j \neq k \\ 0, & j = k \end{cases} \quad (5.8)$$

where e_{jk} represents the number of collaborations that user j and collaborator k have worked together; the range of e_{jk} is $[0, E_{max}]$; $E_{max} (= \arg \max_{j,k} (e_{jk}))$ is the maximum number of collaborations between any user and any collaborator. Specifically, the values of e_{jk} and E_{max} are obtained and calculated from databases on collaboration.

⁶Based on https://en.wikipedia.org/wiki/Netflix_Prize, the Netflix Prize was an open competition for the best collaborative filtering algorithm to predict user ratings for films, based on previous ratings without any other information about the users or films. Its home page is at <https://www.netflixprize.com/>.

⁷Here, $d (\ll m)$ refers to the number of latent factors that characterize users and collaborators.

⁸All ratings in R are in the interval $[0, 1]$.

Notably, $R_{jk}(> 0)$ represents user j 's known rating for collaborator k . $R_{jk}(= 0)$ is user j 's unknown rating, which imply that user j never collaborate with collaborator k . It also suggests that we don't know how user j prefer to collaborate with collaborator k . This require us to predict such ratings by applying PMF. During this process, we need to bound the range of these ratings, since the predicted ratings R_{jk}^* in PMF (see Equation 5.2) may extend beyond the range of known ratings. Thus, we apply the logistic function $\mathcal{G}(U_j^T I_k)$ to replace $U_j^T I_k$ in Equation 5.2 and 5.4 (Mnih and Salakhutdinov, 2008). Finally, users' unknown ratings for collaborators can be predicted by Equation 5.9.

$$R_{jk}^* = \begin{cases} \mathcal{G}(U_j^T I_k), & j \neq k \\ 0, & j = k \end{cases} \quad (5.9)$$

where the user matrix U and the collaborator matrix I are obtained by finding a local minimum of Equation 5.4.

These ratings enables PMF to generate 2D collaborator recommendations without employing any information belonging to the collaboration context (represented as $Context(C)$ in Figure 5.1) in web-based CWE. Besides, as a model-based CF approach technique, PMF has already been widely used, such as generating social recommendations (Ma et al., 2008) and movie recommendations (Yang, Wang, and Chen, 2014). Therefore, we decide to implement PMF in web-based CWEs to generate 2D collaborator recommendations for users.

5.4 Treatment of the collaboration context

This section first presents the formalized terminologies used in the ontology-based context model MCC (cf. Section 4.3). We then explain why the collaboration context can be processed through semantic similarity to generate context-aware collaborator recommendations. Finally, we show how to calculate such a semantic similarity in MCC.

5.4.1 Terminology

As an ontology-based collaboration context model, MCC specifies all information through semantic 3-uples $\langle \text{Subject, Predicate, Object} \rangle$, such as $\langle \text{mcc:Context_aware_application_project, mcc:startedAtTime, "01/07/2018"} \rangle$ representing the start time of Emma's collaboration (as shown in Figure 4.6). In this semantic 3-uple, the subject is an instance of mcc:UserGroup , representing Emma's collaboration. The predicate refers to a contextual factor of collaboration, start time (cf. Section 4.2). Consequently, the object shows the information of Emma's collaboration start time (i.e., the value of the contextual factor represented by the predicate). In this way, for a collaboration in a corresponding user group, all its contextual factors and information can be linked to a single subject in MCC. Particularly, MCC

also allows us to group predicates for expressing the eight contextual dimensions of collaboration: Goal, Collaborator, Activity, Resource, Time, Location, Relation, and Satisfaction (cf. Section 4.2).

Therefore, a collaboration in an instance of `mcc:UserGroup` c is formulated as a collection of semantic 3-uples $\{ \langle c, p_g^{c,t}, o_g^{c,t} \rangle \mid g \leq G_c, t \leq T, g, t \in \mathbb{N}^+ \}$. Here, $p_g^{c,t}$ ($g \leq G_c, t \leq 8, g, t \in \mathbb{N}^+$) is a predicate that the collaboration c contains, which represents a contextual factor in the t^{th} dimension; $P^{c,t} = \{ p_g^{c,t} \mid g \leq G_c, g, t \in \mathbb{N}^+ \}$ ($t = 1, 2, \dots, T$) is a set of predicates that the collaboration c contains in the t^{th} contextual dimension; $P^c = \{ p_g^c \mid g \leq G_c, g \in \mathbb{N}^+ \}$ is the set of all predicates that the collaboration c contains; $o_g^{c,t} \in O^{c,t}$ is an object of the predicates $p_g^{c,t}$, which provides value to the corresponding contextual factor in the t^{th} dimension; $O^{c,t} = \{ o_g^{c,t} \mid g \leq G_c, t \leq 8, g, t \in \mathbb{N}^+ \}$ is a set of objects that the collaboration c contains through predicates in $P^{c,t}$; $O^c = \{ o_g^c \mid g \leq G_c, g \in \mathbb{N}^+ \}$ is the set of all objects that the collaboration c contains through predicates in P^c ; G_c represents the number of predicates (contextual factors) that the collaboration c contain; $T (= 8)$ represents the maximum number of contextual dimensions that the collaboration c can relate⁹.

Accordingly, the t^{th} contextual dimension of the collaboration c can be considered a sub-collection $\{ \langle c, p_g^{c,t}, o_g^{c,t} \rangle \mid p_g^{c,t} \in P^{c,t}, o_g^{c,t} \in O^{c,t} \}$. For example, the contextual dimension *collaborator* is represented as $\{ \langle c, p_g^{c,Col}, o_g^{c,Col} \rangle \mid p_g^{c,Col} \in P^{c,Col}, o_g^{c,Col} \in O^{c,Col} \}$. This implies that collaborations and their contextual dimensions¹⁰ can be described and interpreted as collections of semantic 3-uples in MCC.

Specifically, in these collections of semantic 3-uples, there are two types of predicates (as shown in Figure 4.3). The first type are directly related to `mcc:UserGroup`, such as datatype properties (e.g. `mcc:startedAtTime`) and object properties (e.g. `mcc:isRelatedWith`). The other type of predicates are represented as successions of datatype properties, object properties, associated classes and/or instances. Such predicates¹¹ are indirectly related to `mcc:UserGroup` in MCC. The succession of `mcc:hasGoal`, `mcc:Goal`, and `mcc:isComposedOf` is an example.

Besides, in a collection of semantic 3-uples, the subject may be other instances in MCC. For example, the resource `kb:Scientific_paper` created by Emma can be considered as a subject (see Figure 4.11). Then the collection of semantic 3-uples related to this particular subject describes the detailed information of this resource, such as $\langle \text{kb:Scientific_paper}, \text{rdf:type}, \text{mc2:Document} \rangle$ representing the type of this resource.

⁹When $t = T$, the collaboration c contains information of all the eight contextual dimensions in MCC, as follows: Goal, Collaborator, Activity, Resource, Time, Location, Relation, and Satisfaction.

¹⁰In MCC, the contextual dimensions of a collaboration include Goal, Collaborator, Activity, Resource, Time, Location, Relation, and Satisfaction.

¹¹A summary of the indirect predicates in the five dimensions (Goal, Collaborator, Activity, and Resource) are available in Figure 5.2.

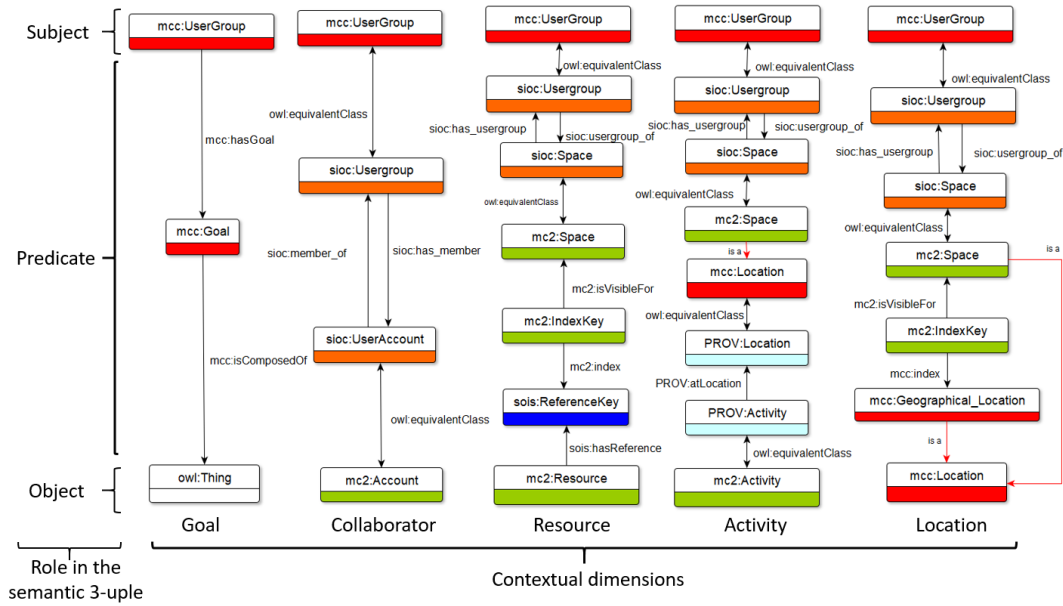


FIGURE 5.2: Distribution of classes and interrelations according to their role in the semantic 3-uple.

5.4.2 Choosing semantic similarity

While processing the collaboration context, there are many options, such as ontology-based reasoning (Dou, Wang, and Liu, 2015), ontology-based semantic similarity (Sánchez et al., 2012; Batet, Sánchez, and Valls, 2011; Zhang et al., 2008; Ovaska, Laakso, and Hautaniemi, 2008). Among them, we choose to compute semantic similarities between collaborations for two reasons.

1. Unlike reasoning, semantic similarities are not overly dependent on the definitions of classes, instances and their relationships in ontologies (Roussey et al., 2011). For instance, one type semantic similarity (i.e., feature-based semantic similarity) can even be used to measure similarities between classes (instances) in different ontologies (Petrakis et al., 2006). This makes it easier for us to define classes, instances, and their relationships in MCC, describing heterogeneous information from different collaborative tools in web-based CWEs.
2. In ontologies, semantic similarities enable comparison of classes (instances) and/or their relationships at different levels. For example, in MCC, semantic similarities can be used not only to compare two user groups (i.e., instances of `mcc:UserGroup`) or resources (i.e., instances of `mc2:Resource`), but also to measure the similarity between indexed resources within a user group. This implies that calculating semantic similarities in MCC enable us to compare any two classes (instances) and/or their relationships, involved in the collaboration context.

Therefore, the collaboration context is processed through semantic similarities in this thesis. This gives us the possibility to filter and/or adjust users' ratings in generating context-aware collaborator recommendations, following PreF and PoF methods (see Figure 5.1).

5.4.3 Computing semantic similarity

Based on the literature review (cf. Section 5.2.2), existing ontology-based semantic similarities can be classified into three types (Sánchez et al., 2012; Batet, Sánchez, and Valls, 2011; Zhang et al., 2008; Ovaska, Laakso, and Hautaniemi, 2008): path-based, feature-based, and information content-based semantic similarity. These different semantic similarities have their own disadvantages and advantages. Therefore, we need to compute semantic similarities based on the characteristics of our ontology-based collaboration context model, MCC.

MCC is a core ontology used by different users and user groups in web-based CWEs (Roussey et al., 2011). Meanwhile, it also serves as an ontological knowledge base for storing and manipulating information across collaborative tools integrated into a web-based CWE, which will be presented and explained in Chapter 4. Thus, the relationships in MCC don't represent uniform distances between classes and instances. This prevents us from calculating path-based semantic similarities in MCC, because of its dependency on the relationships in an ontology (Sánchez et al., 2012).

Besides, in MCC, a single subject can relate to different predicates and objects within a collection of semantic 3-uples. Even with the same predicate, objects linked to a subject can be varied. For example, two members in Emma's collaboration (see Figure 4.6) are represented by `<mcc:Context_aware_application_project, mcc:has_member, #Lucie_account>` and `<mcc:Context_aware_application_project, mcc:has_member, #Marinela_account>`.

Therefore, the semantic similarity in MCC is expected not only to compare the common and non-common objects in two collections of semantic 3-uples `<Subject, Predicate, Object>`, but also to assess the amount of information provided by a specific predicate. To this end, a new semantic similarity is developed and employed in MCC, inspired by feature-based and information context-based semantic similarities. Such a semantic similarity compares two collections of semantic 3-uples (describing two subjects x and y) from two aspects: object (built on feature-based semantic similarities) and predicate (built on information content-based semantic similarities).

5.4.3.1 Object

Drawing inspiration from feature-based semantic similarities, this aspect measures the common and non-common objects that are related to the same predicate in two collections of

semantic 3-uples. Depending on the object type in a collection, both qualitative and quantitative objects are considered.

5.4.3.1.1 Qualitative object

Qualitative objects contain non-numerical and descriptive information. For example, when comparing whether there are same collaborators in two collections, collaborators are qualitative objects. Such objects that are linked to the same predicate p_g^t in two collections of semantic 3-uples constitute two sets: $O_{p_g^t}^x$ and $O_{p_g^t}^y$. These sets can be measures by the following equation:

$$S_1(x, y) = \sum_{t=1}^{T_{qual}} \sum_{g=1}^{|P^{xy,t}|} \frac{|O_{p_g^t}^x \cap O_{p_g^t}^y|}{|O_{p_g^t}^x \cap O_{p_g^t}^y| + \alpha |O_{p_g^t}^x - O_{p_g^t}^y| + \beta |O_{p_g^t}^y - O_{p_g^t}^x|} \times \frac{IC(P^{x,t})}{\sum_{h=1}^T IC(P^{x,h})} \quad (5.10)$$

where T_{qual} ($T_{qual} \leq T, T_{quan} + T_{qual} = T, T_{qual} \in \mathbb{N}^+$) is the number of contextual dimensions that include qualitative objects; $O_{p_g^t}^y$ indicates a set of objects that y relates through the predicate p_g^t ; $O_{p_g^t}^x - O_{p_g^t}^y$ denotes the relative complement of $O_{p_g^t}^y$ in $O_{p_g^t}^x$; $|O_{p_g^t}^x \cap O_{p_g^t}^y|$ represents the number of common objects associated to x and y through the predicate p_g^t ; $|P^{xy,t}|$ represents the number of predicate types belonging to the t^{th} contextual dimension of both x and y ; $IC(P^{x,t}) (\geq 0)$ expresses the amount of information provided by the predicates in the t^{th} contextual dimension of the collaboration x ($t = 1, 2, \dots, T$)¹²; $\sum_{h=1}^T IC(P^{x,h})$ is the sum amount of information provided by all the contextual dimensions of the collaboration x ; $\alpha, \beta \geq 0$.

The range of $S_1(x, y)$ is $[0, \frac{\sum_{t=1}^{T_{qual}} IC(P^{x,t})}{\sum_{h=1}^T IC(P^{x,h})}]$. Smaller the semantic similarity $S_1(x, y)$, more differences between x and y .

5.4.3.1.2 Quantitative object

Quantitative objects include numerical information, such as start time and end time of a collaboration (cf. Section 4.2). To compare two such objects, we utilize the absolute difference between them $|o_{p_g^t}^x - o_{p_g^t}^y|$. The smaller the absolute difference is, the greater the similarity between $o_{p_g^t}^x$ and $o_{p_g^t}^y$. Thus, we apply Equation 5.11 to convert $|o_{p_g^t}^x - o_{p_g^t}^y|$.¹³

$$S_2(x, y) = \sum_{l=1}^{T_{quan}} \sum_{g=1}^{|P^{xy,t}|} \frac{1}{\gamma |o_{p_g^t}^x - o_{p_g^t}^y| + 1} \times \frac{IC(P^{x,t})}{\sum_{h=1}^T IC(P^{x,h})} \quad (5.11)$$

¹²The calculation of $IC(P^{x,t})$ will be presented in Section 5.4.3.2.

¹³Equation 5.11 is inspired from the above equation of qualitative semantic similarity and the discussion on the site <https://stats.stackexchange.com/questions/158279/how-i-can-convert-distance-euclidean-to-similarity-score>. In Equation 5.11, we only concern about the predicates linking to limited objects.

where T_{quan} ($T_{quan} \leq T, T_{quan} + T_{qual} = T, T_{quan} \in \mathbb{N}^+$) is the number of contextual dimensions that include quantitative objects; $o_{p_g^t}^x$ indicates a quantitative object that x relates p_g^t ; $|o_{p_g^t}^x - o_{p_g^t}^y|$ denotes the absolute differences between two objects of x and y related to the predicate p_g^t ; $|P^{xy,t}|$ represents the number of predicate types belonging to the t^{th} contextual dimension that both x and y relate; $IC(P^{x,t}) (\geq 0)$ expresses the amount of information provided by the predicates in the t^{th} contextual dimension of the collaboration x ($t = 1, 2, \dots, T$)¹⁴; $\sum_{h=1}^T IC(P^{x,h})$ is the sum amount of information provided by contextual dimensions of the collaboration x ; $\gamma \geq 0$.

The range of $S_2(x, y)$ is $[0, \frac{\sum_{t=1}^{T_{quan}} IC(P^{x,t})}{\sum_{h=1}^T IC(P^{x,h})}]$. Besides, $S_2(x, y) \neq S_2(y, x)$ due to the different values of $IC(P^{x,t})$ and $IC(P^{y,t})$. Notably, Equation 5.11 is suitable for $o_{p_g^t}$ whose range of is $[0, +\infty)$. Particularly, if the range of $o_{p_g^t}$ is $[0, L]$, then the above equation is transformed into:

$$S_2(x, y) = \sum_{t=1}^{T_{quan}} \sum_{g=1}^{|P^{xy,t}|} \left(\frac{\frac{L+1}{L}}{\gamma |o_{p_g^t}^x - o_{p_g^t}^y| + 1} - \frac{1}{L} \right) \times \frac{IC(P^{x,t})}{\sum_{h=1}^T IC(P^{x,h})} \quad (5.12)$$

5.4.3.1.3 All object

Combining Equation 5.10, 5.11, and 5.12, we can obtain the semantic similarity between x and y in the object aspect (see Equation 5.13).

$$S(x, y) = S_1(x, y) + S_2(x, y) \quad (5.13)$$

The range of $S(x, y)$ is $[0, \frac{\sum_{i=1}^{T_{quan}} IC(P^{x,i}) + \sum_{i=1}^{T_{qual}} IC(P^{x,i})}{\sum_{h=1}^T IC(P^{x,h})}]$. With $T_{quan} + T_{qual} = T$, we have $S(x, y) \in [0, 1]$. Smaller semantic similarity $S(x, y)$ implies greater difference between x and y .

5.4.3.2 Predicate

This aspect accesses the amount of information provided by a specific predicate in collections of semantic 3-uples, inspired by information content-based semantic similarities. Particularly, a predicate can relate to one subject multiple times with different objects in a collection. This indicates that the appearance frequency of a predicate depends on the collections, affecting the amount of information provided by this predicate. Therefore, the appearance frequency of a predicate should be considered within the semantic similarity. Besides, the contribution (i.e., how important a predicate is to a subject (Rajaraman and Ullman, 2011)) of a predicate to a collection can also influence the amount of information provided by this

¹⁴The calculation of $IC(P^{x,t})$ will be presented in Section 5.4.3.2.

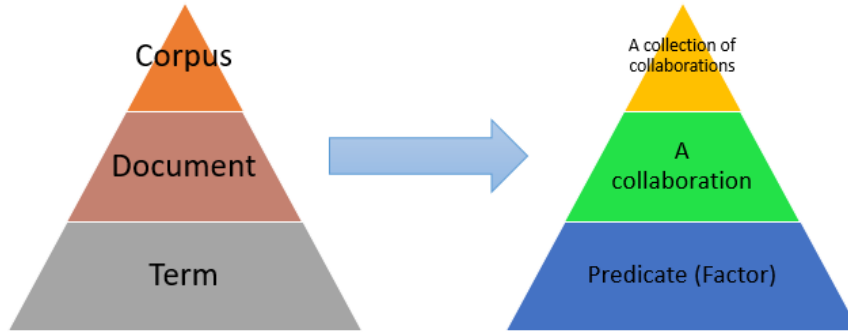


FIGURE 5.3: Apply TF-IDF to measure the amount of information provided by predicates.

predicate. To consider both the appearance frequency and the contribution of a predicate, TF-IDF is applied to measure the amount of information provided by a predicate. Specifically, we replace “term, document, and corpus” in TF-IDF as following for “predicate (factor), a collaboration, and a collection of collaborations” (see Figure 5.3).

For a predicate p in a collaboration x ($x \in X$, X represents a collection of collaborations) and the total number of collaborations in the collection is $|X|$, we have

$$\begin{aligned} tf(p, x) &= \frac{f(p, P^x)}{\sum_{p' \in P^x} f(p', P^x)} \\ idf(p, X) &= \log \frac{|X|}{1 + |\{x \in X : p \in P^x\}|} + 1 \\ tf \cdot idf(p, x, X) &= tf(p, x) \cdot idf(p, X) \end{aligned} \quad (5.14)$$

where $f(p, P^x)$ represents the appearance frequency of the predicate p in the collaboration x ; $\sum_{p' \in P^x} f(p', P^x)$ denotes the appearance frequencies of all predicates in the collaboration x . The range¹⁵ of $tf \cdot idf(p, x, X)$ is $[0, \log|X| + 1]$.

Then we can get the amount of information provided by the predicate p in the collaboration x , $IC(p) = tf \cdot idf(p, x, X)$. As for the amount of information provided by the t^{th} dimension: $IC(P^{x,t})(t = 1, 2, \dots, T)$, it is calculated through Equation 5.15:

$$IC(P^{x,t}) = \sum_{q=1}^{|P^{x,t}|} tf \cdot idf(p_q^{x,t}, x, X) \quad (5.15)$$

where $|P^{x,t}|$ is the number of predicate types in the t^{th} dimension of the collaboration x ; $p_q^{x,t} \in P^{x,t}$. The range of $IC(P^{x,t})$ is $[0, 2(\log|X| + 1)]$.

Equation 5.15 is used in Equation 5.10, 5.11, and 5.12 to compute the ontology-based semantic similarity between two collections of semantic 3-uples.

¹⁵Particularly, the range of $tf(p, x)$ is $[0, 1]$ and the range of $idf(p, X)$ is $[\log \frac{|X|}{1+|X|} + 1, \log|X| + 1]$.

5.4.3.3 Calculating semantic similarity in MCC

In MCC, we have $T = 8$ and $T_{quan} = 6$, including Goal, Collaborator, Activity, Resource, Location, and Relation (cf. Section 4.2). Thus, based on Equation 5.10, we have:

$$S_1(x, y) = \sum_{t_1=1}^6 \sum_{g_1=1}^{|P^{xy,t_1}|} \frac{|O_{p_{g_1}^{t_1}}^x \cap O_{p_{g_1}^{t_1}}^y|}{|O_{p_{g_1}^{t_1}}^x \cap O_{p_{g_1}^{t_1}}^y| + \alpha |O_{p_{g_1}^{t_1}}^x - O_{p_{g_1}^{t_1}}^y| + \beta |O_{p_{g_1}^{t_1}}^y - O_{p_{g_1}^{t_1}}^x|} \times \frac{IC(P^{x,t_1})}{\sum_{h=1}^8 IC(P^{x,h})} \quad (5.16)$$

where $|P^{xy,t_1}|$ is the number of predicate types belonging to the t_1^{th} dimension that both x and y relate; $\alpha, \beta \geq 0$; $IC(P^{x,t_1})$ is calculated from Equation 5.15; $\sum_{h=1}^8 IC(P^{x,h})$ is the sum amount of information provided by the predicates in all contextual dimensions of the collaboration x .

Besides, the rest two dimensions (Time and Satisfaction) are utilizing quantitative objects (cf. Section 4.2). Particularly, the range of objects in dimension Time is $[0, +\infty)$, while the range of objects in dimension Satisfaction is $[0, L_{Sa}]$. Thus, based on Equation 5.11 and 5.12, we can obtain:

$$S_2(x, y) = \sum_{g_2=1}^{|P^{xy,Time}|} \frac{1}{\gamma_1 |o_{p_{g_2}^{Time}}^x - o_{p_{g_2}^{Time}}^y| + 1} \times \frac{IC(P^{x,Time})}{\sum_{h=1}^8 IC(P^{x,h})} + \left(\frac{\frac{L_{Sa}+1}{L_{Sa}}}{\gamma_2 |\bar{o}_{p_{Sa}}^x - \bar{o}_{p_{Sa}}^y| + 1} - \frac{1}{L_{Sa}} \right) \times \frac{IC(P^{x,Sa})}{\sum_{h=1}^8 IC(P^{x,h})} \quad (5.17)$$

where $|P^{xy,Time}|$ is the number of predicate types belonging to the dimension Time that both x and y relate; L_{Sa} is the maximum value in the range of individuals' satisfactions to collaborations¹⁶; $\bar{o}_{p_{Sa}}^x$ denotes the average of all given satisfactions to the collaboration x ; $\gamma_1, \gamma_2 \geq 0$; $IC(P^{x,Time})$ and $IC(P^{x,Sa})$ are also calculated from Equation 5.15.

Combining Equation 5.16 and 5.17, the semantic similarity between x and y in MCC is:

$$\begin{aligned} S(x, y) &= S_1(x, y) + S_2(x, y) \\ &= \sum_{t_1=1}^6 \sum_{g_1=1}^{|P^{xy,t_1}|} \frac{|O_{p_{g_1}^{t_1}}^x \cap O_{p_{g_1}^{t_1}}^y|}{|O_{p_{g_1}^{t_1}}^x \cap O_{p_{g_1}^{t_1}}^y| + \alpha |O_{p_{g_1}^{t_1}}^x - O_{p_{g_1}^{t_1}}^y| + \beta |O_{p_{g_1}^{t_1}}^y - O_{p_{g_1}^{t_1}}^x|} \times \frac{IC(P^{x,t_1})}{\sum_{h=1}^8 IC(P^{x,h})} \\ &+ \sum_{g_2=1}^{|P^{xy,Time}|} \frac{1}{\gamma_1 |o_{p_{g_2}^{Time}}^x - o_{p_{g_2}^{Time}}^y| + 1} \times \frac{IC(P^{x,Time})}{\sum_{h=1}^8 IC(P^{x,h})} \\ &+ \left(\frac{\frac{L_{Sa}+1}{L_{Sa}}}{\gamma_2 |\bar{o}_{p_{Sa}}^x - \bar{o}_{p_{Sa}}^y| + 1} - \frac{1}{L_{Sa}} \right) \times \frac{IC(P^{x,Sa})}{\sum_{h=1}^8 IC(P^{x,h})} \end{aligned} \quad (5.18)$$

where the range of $S(x, y)$ is $[0, 1]$.

¹⁶For example, if individuals' satisfactions to a collaboration belong to the interval $[0, 5]$, then $L_{Sa} = 5$.

Within this semantic similarity $S(x, y)$, the likeness/relatedness between collections of semantic 3-uples with unique subjects can be measured. When the subjects in the two collections are instances of `mcc:UserGroup` in MCC, a semantic similarity then compares two collaborations and their contexts. In this way, the collaboration context can be processed for generating context-aware collaborator recommendations.

Besides, when the subjects in the two collections of semantic 3-uples are other instances in MCC, the semantic similarity $S(x, y)$ (see Equation 5.13) can also serve to compare the similarities between them. For instance, when x and y represent two resources, then $S(x, y)$ can be calculated based on the objects and predicates in their collections, including identifying common and non-common objects and determining the amount of information provided by the predicates.

5.5 Context-aware collaborator recommendation algorithms

Within the collections of semantic 3-uples and the proposed semantic similarity in MCC (cf. Section 5.4), the context-aware collaborator recommendation problem is formulated as:

Given a utility matrix R and a user u in a collaboration c ($u \in O^{c, Col}$), the top K collaborators i ($i \neq u, i \notin O^{c, Col}$) that can facilitate u 's collaboration c in a corresponding user group with the highest probabilities will be recommended to u . Here, the context indicates the context of u 's collaboration c , which is represented by a collection of semantic 3-uples $\{ \langle c, p_g^{c,t}, o_g^{c,t} \rangle \mid g \leq G_c, t \leq 8, g, t \in \mathbb{N}^+ \}$.

Specifically, we apply two methods: PreF and PoF, to generate context-aware collaborator recommendations. Based on these methods, three algorithms are developed, known as PreF1, PoF1, and PoF2 in Algorithm 1). Some specific functions are presented in Algorithms 2 and 3.

Particularly, when $z = 0$, Algorithm 1 refers to PreF1 algorithm, composed of three phases: Phase 1, 2, and 4 (see Algorithm 1). It first calculates ontology-based semantic similarities between the collaboration c and other collaborations d ($d \in X, d \neq c$) (line 1), based on Equation 5.18. Then it filters out irrelevant collaborators i' that did not participate in similar collaborations with c (line 2-3). Next, PMF is applied to predict users' unknown ratings for relevant collaborators (line 4-8). Finally, the top K collaborators with higher ratings will be recommended to the user u (line 27). Briefly, following PreF method (see Figure 5.1), PreF1 algorithm first pre-processes the collaboration context by means of an ontology-based semantic similarity, then generates 2D recommendations by using PMF, and finally produce context-aware collaborator recommendations.

Besides, PoF1 algorithm has $z = 1$. It also consists of three phases: Phase 2, 3, and 4 (see Algorithm 1). In PoF1, we first utilize PMF to predict users' unknown ratings for collaborators (line 9-12). Then, it computes ontology-based semantic similarities between the

Algorithm 1: PreF1, PoF1, and PoF2 algorithms.

Input: The rating matrix: R ,
the user to whom the recommendations are generated: u ,
the collaboration: c ,
the set of members in the collaboration c : O_{Col}^c ,
the number of recommendations: K ,
the number of known collaborations: $|X|$,
the number of users that can be recommended: m ,
the type of algorithms: z ,
the weight of semantic similarity in adjusted ratings: w_s ,
the weight of predicted rating in adjusted ratings: w_r .

Initialization: A list to save predicted ratings: \hat{R} (length: m , initial values: 0).

Comment: Phase 1 - Pre-processing the collaboration context.

1 $SC_0 \leftarrow Calculation_SS(c, K, |X|, 1)$;
2 $VU_0 \leftarrow Filter(R, u, SC_0, m, 1)$;
3 Filter R to R' by deleting all ratings of irrelevant collaborators i' ($VU_0(i') = 0$);

Comment: Phase 2 - Generating 2D collaborator recommendations.

4 **if** $z == 0$ **then**
5 **for** $k \in \{1, 2, \dots, m\}$ **do**
6 **if** R'_{uk} is unknown **then**
7 apply PMF to predict $\hat{R}(k) \leftarrow VU_0(k) \times R'_{jk}$ based on Equation 5.9;
8 **go to** line 27;
9 **else**
10 **for** $k \in \{1, 2, \dots, m\}$ **do**
11 **if** R_{uk} is unknown **then**
12 apply PMF to predict $R_{uk} \leftarrow R_{uk}^*$ based on Equation 5.9;

Comment: Phase 3 - Post-processing the collaboration context.

13 $SC_1 \leftarrow Calculation_SS(c, K, |X|, 2)$;
14 $VU_1 \leftarrow Filter(R, u, SC_1, m, 3)$;
15 $SC_2 \leftarrow Calculation_SS(c, K, |X|, 1)$;
16 $VU_2 \leftarrow Filter(R, u, SC_2, m, 2)$;
17 **if** $z == 1$ **then**
18 **for** $k \in \{1, 2, \dots, m\}$ **do**
19 **if** $k \notin O^{c, Col}$ **then**
20 **if** $VU_1(k) == 1$ **then**
21 $\hat{R}(k) \leftarrow w_s * sum(SC_1(k)) + w_r * R_{uk}$;
22 **go to** line 27;
23 **if** $z == 2$ **then**
24 **for** $k \in \{1, 2, \dots, m\}$ **do**
25 $\hat{R}(k) \leftarrow R_{uk} \times VU_2(k)$

Comment: Phase 4 - Producing context-aware collaborator recommendations.

26 Rank \hat{R} in decreasing order and get K highest elements;

Output: User u 's predicted ratings \hat{R} , K collaborators with K highest ratings.

Algorithm 2: Calculating semantic similarity function.

Input: The collaboration: c ,
the number of recommendations: K ,
the number of collaborations: $|X|$,
the type of results: res_1 .

Initialization:

Two lists to save semantic similarities: SC_1, SC_2 (length: $|X|$, initial values: 0).

```

1 Function Calculation_SS ( $c, K, |X|, res_{ss}$ ):
2   for  $d \in \{1, 2, \dots, |X|\}$  do
3     if  $d == c$  then
4        $S(d, c) \leftarrow 0$ ;
5     else
6        $S(d, c) \leftarrow S_1(d, c) + S_2(d, c)$  based on Equation 5.18;
7        $SC_1.insert(S(d, c); d)$ ;
8       while  $i \in O_{Col}^d$  do
9         if collaboration  $d$  belongs to collaborator  $i$ 's recent  $K$  collaborations
10        then
11           $SC_2(i).insert(S(d, c))$ ;
12   Rank  $SC_1$  in decreasing order based on the first value of each element and copy
13    $K$  highest elements' second values in a new list  $SC'_1$ ;
14   if  $res_1 == 1$  then
15      $SC \leftarrow SC'_1$ ;
16   else
17      $SC \leftarrow SC_2$ ;
18   return  $SC$ ;

```

Output: A list of semantic similarities SC .

Algorithm 3: Filtering irrelevant collaborator function.**Input:** The rating matrix: R ,the user to whom the recommendations are generated: u ,a list of semantic similarities: SC ,the number of users that can be recommended: m ,the type of results: res_2 .**Initialization:**Three lists to identify relevant collaborators: VU_1, VU_2, VU_3 (length: m , initial values: 0).

```

1 Function Filter( $R, u, SC, m, res_2$ ):
2   for  $k \in \{1, 2, \dots, m\}$  do
3     if  $R_{uk}$  is unknown then
4        $VU_1(k) \leftarrow 1$ ;
5        $VU_3(k) \leftarrow 1$ ;
6   for  $k \in \{1, 2, \dots, m\}$  do
7     for  $d \in \{1, 2, \dots, length(SC)\}$  do
8       if  $k \in O_{Col}^d$  then
9          $VU_1(k) \leftarrow 1$ ;
10         $VU_2(k) \leftarrow 1$ ;
11   if  $res_2 == 1$  then
12      $VU \leftarrow VU_1$ ;
13   else
14     if  $res_2 == 2$  then
15        $VU \leftarrow VU_2$ ;
16     else
17        $VU \leftarrow VU_3$ ;
18   return  $VU$ ;

```

Output: A list to identify relevant collaborators VU .

collaboration c and other collaborations d ($d \in X, d \neq c$), and a sum of semantic similarities between the collaboration c and the recent K collaborations that collaborator i ($i \neq u$) participated (line 13-14). Next, users' adjusted ratings for collaborators are calculated based on predicted ratings, calculated semantic similarities, and their weights w_s, w_r (line 17-22). Finally, the top K collaborators with higher adjusted ratings will be recommended to the user u (line 27).

Finally, when $z = 2$ in Algorithm 1, PoF2 algorithm can be used through three phases: Phase 2, 3, and 4. It first employs PMF to predict users' unknown ratings for collaborators (line 9-12). Then, the ontology-based semantic similarities between the collaboration c and other collaborations d ($d \in X, d \neq c$) is computed (line 15). Next, it filters out irrelevant collaborators i' that did not participate in similar collaborations with c (line 16). Finally, the top K collaborators with higher ratings will be recommended to the user u (line 23-27).

The differences between PoF1 and PoF2 algorithms are on how to apply semantic similarity in filtering or adjusting the the order of collaborators i ($i \neq u, i \notin O^{c,Col}$). The first PoF algorithm (i.e., $z = 1$) utilizes the K recent collaborations that collaborators i participated, while the other (i.e., $z = 2$) employs the collaborations with K highest semantic similarity among all $|X|$ collaborations. However, to generate context-aware collaborator recommendations in PoF method (see Figure 5.1), both PoF1 and PoF2 generate 2D collaborator recommendations by using PMF, then process the collaboration context using semantic similarity, and eventually produce context-aware collaborator recommendations.

Using the three algorithms: PreF1, PoF1, and PoF2 (see Algorithm 1), context-aware collaborator recommendations can be produced for users in web-based CWEs. Specifically, all these algorithms employ PMF and the ontology-based semantic similarities in the recommendation generating process, which makes it possible to compare and evaluate their performances through experiments. These experiments will be presented and explained in Chapter 7.

5.6 Chapter summary

In this chapter, we concentrated on utilizing the collaboration context to generate context-aware collaborator recommendations. This part of work addresses the second question listed in Section 1.1.3: how to process and apply the collaboration context in algorithms for generating context-aware collaborator recommendations.

Specifically, we chose a 2D recommendation technique, PMF, to produce 2D collaborator recommendations. Then, we decided to handle the collaboration context through a new ontology-based semantic similarity. It allows us to measure the likeness/relatedness between two collections of semantic 3-uples in the ontology-based collaboration context model. Each

collection has a single subject, representing a collaboration and its context. Finally, we explained how to utilize PMF and the ontology-based semantic similarity in three different algorithms to generate context-aware collaborator recommendations.

The next chapter will present the answer to the third question: how to implement and integrate the ontology-based collaboration context model into web-based CWEs.

Chapter 6

Prototype

6.1 Introduction

Through the ontology-based model MCC (cf. Chapter 4), the context of users' collaborations in web-based Collaborative Working Environments (CWEs) can be specified and represented. Then the next question we need to solve is: how to implement and integrate MCC into web-based CWEs? For this, it is necessary to explore how to deploy MCC from an architectural perspective. Meantime, since a web-based CWE is a collaborative System of Information Systems (SoIS), the architectures of collaborative SoIS equally are applicable to web-based CWEs. Therefore, we consider the architecture of an ontology-based collaborative SoIS (Saleh and Abel, 2016) from the viewpoint of web-based CWEs and employ MCC as a shared ontological knowledge base in web-based CWEs.

Based on this architecture, a corresponding prototype of web-based CWEs can be built. The prototype is a continuation of the MEMORAe approach (Atrash, Abel, and Moulin, 2014), known as MEMORAe CWE. It is capable of not only supporting users' collaborations across multiple collaborative tools, but also processing and organizing heterogeneous information from these tools within the collaboration context. Specifically, such information is managed in the knowledge base, MCC.

The remainder of this chapter is organized as follows: Section 6.2 presents how to consider the architecture of the ontology-based collaborative SoIS from the viewpoint of web-based CWEs and employ MCC in it. Then, the web-based CWE prototype, MEMORAe CWE, is presented in Section 6.3. We also discuss its advantages and disadvantages in Section 6.4. At the end of this chapter is the summary.

6.2 From an architecture of collaborative System of Information Systems towards an architecture of web-based Collaborative Working Environments

Saleh and Abel (2016) proposed a leader-follower architecture of collaborative SoISs (see Part (a) in Figure 6.1). In this architecture, a SoIS is composed of a leader system that refers to the global system and multiple follower systems that are component systems. Particularly, the global system contains an ontological knowledge base to collectively manage heterogeneous information across different information systems.

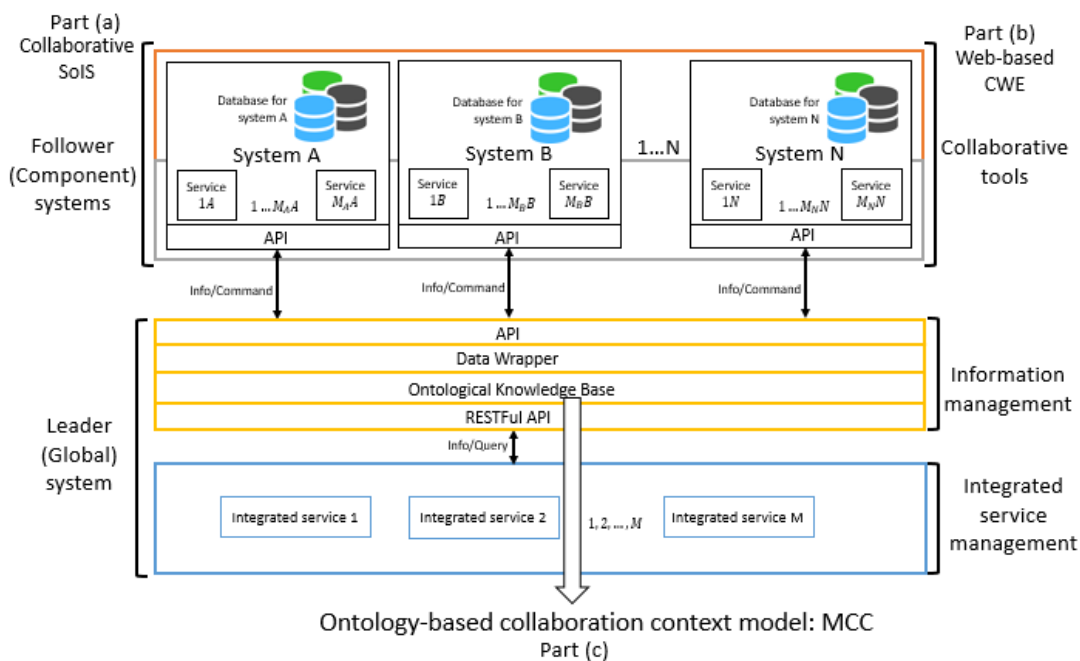


FIGURE 6.1: Considering a web-based CWE as a collaborative SoIS.

Considering a web-based CWE as a collaborative SoIS, this leader-follower architecture can also serve as an architecture of web-based CWEs (see Part (b) in Figure 6.1). Specifically, the component systems are collaborative tools that provide independent services (e.g., service 1A, ..., $M_A A$ in Figure 6.1) and operate under their own control. Moreover, the global system indicates a web-based CWE with integrated services (e.g., integrated service 1, ..., M in Figure 6.1), which are results of collaborations between the integrated tools. Particularly, collaborative tools are added in a web-based CWE to offer and integrate their services into the CWE, thus supporting users' collaborations. When they are not desired, they can also be removed from a web-based CWE by disconnecting with the CWE.

Besides, four parts can manage heterogeneous information in the web-based CWE: *API*, *data wrapper*, *ontological knowledge base*, and *RESTful API*. *API* exchanges information with multiple collaborative tools. Then, *data wrappers* aims to manage the information collected through *API* and to deploy it into different instances in an *ontological knowledge base*.

Finally, *RESTful API* offers uniform interfaces¹ for accessing such instances and manipulating their state (Lucchi, Millot, and Elfers, 2008). Such an architecture (see Part (b) in Figure 6.1) allows heterogeneous information located in independent tools to be visible and accessible in web-based CWEs without modifying its storage location (source).

Particularly, thanks to the ontological knowledge base, information collected from different tools can be managed in a single expressive manner. Such a knowledge base can represent and map heterogeneous information into semantic instances, classes, and relationships. This permits the web-based CWE to have a uniform representation of heterogeneous information, regardless of where it is contained. In this way, the interoperability between the collaborative tools integrated in a web-based CWE can be improved.

However, web-based CWEs under this architecture (see Part (b) in Figure 6.1) do not consider the collaboration context (e.g., users' activities, applied resources) when organizing and retrieving the information to meet users' needs during collaborations, though these needs are influenced by the collaboration context. This implies that the information contained in web-based CWEs should be managed within the collaboration context. Thus, we implement the ontology-based collaboration context model MCC as the ontological knowledge base in the architecture of web-based CWEs (see Part (c) in Figure 6.1). In this way, any information belonging to the collaboration context becomes directly accessible components in MCC. Here, a component indicates an instance (e.g. instances of `mcc:UserGroup`, `mc2:Activity`) in MCC, whose unique identification is represented by URIs. Specifically, each instance is described by its characteristic information and can be accessed in JSON format through *RESTful API*. For example, below shows the detailed information of an instance of `mcc:UserGroup`.

```
1 {"id": "groupAPIConcept5fa51d8caea88",
2  "name": "Context-aware application project",
3  "member": [
4    {"login": "Emma_account", "id": "testAPIConcept5db5b8e6250f8"},
5    {"login": "Marinela_account", "id": "testAPIConcept5dc56c6943da5"},
6    {"login": "Lucie_account", "id": "testAPIConcept5db5bbbac5405"}
7  ],
8  "type": "Project Group",
9  "goal": "Build a context-aware application and publish a
10         corresponding paper",
11  "starttime": "01-11-2020",
12  "endtime": "31-12-2021",
   "related_group": [
```

¹Particularly, for accessing and manipulating the instances, two different interfaces are provided separately to the instances (see Table B.1 in Appendix B) and the lists of instances (see Table B.2 in Appendix B).

```

13 {"name": "Emma's thesis", "id": "groupAPIConcept5fa51f1c04761"
14   },
15 "space": "spaceAPIConcept5fa51d8cbd524"
16 }

```

Through applying MCC as the ontological knowledge base, the collaboration context can be processed and managed in web-based CWEs. This allows users to analyze and identify the weak and strong points in their collaborations, then set up corresponding strategies. Moreover, the leader-follower architecture (see Part (a) in Figure 6.1) was already used to implement a prototype of collaborative SoIS, *MEMORAe SoIS* (Saleh and Abel, 2016), where the ontology *SOIS* acts as the knowledge base (cf. Section 3.3). This implies that the architecture is practicable. Consequently, the architecture of web-based CWEs (see Part (b) and (c) of figure 6.1) leads to the construction of a web-based CWE prototype.

6.3 Prototype: MEMORAe CWE

Following the architecture of web-based CWEs (cf. Section 6.2), we build a corresponding prototype, *MEMORAe CWE*². This prototype is a continuation of the MEMORAe approach (Atrash, Abel, and Moulin, 2014). As presented in Chapter 3, *MEMORAe CWE* is the web platform in the MEMORAe approach, while the ontology-based collaboration context model *MCC* is applied as the core ontology.

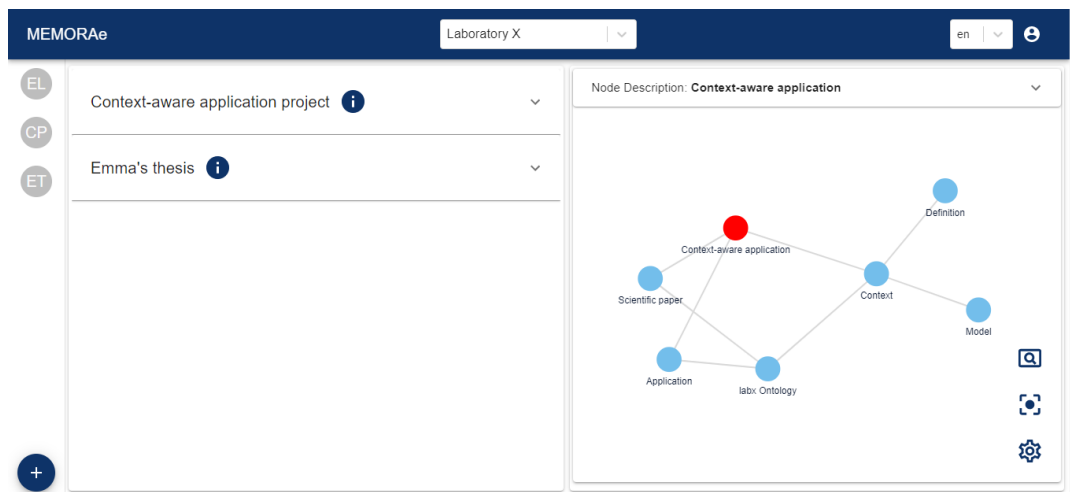


FIGURE 6.2: The main interface of MEMORAe CWE.

Particularly, users in MEMORAe CWE are organized by user groups and organizations. An organization consists of multiple users belonging to interrelated user groups. Moreover, each organization has its own organizational ontology where users are allowed to index and

²A demonstration video for MEMORAe CWE is available at <https://youtu.be/c3xFMI1bktk>.

manage resources within the organization (Atrash, Abel, and Moulin, 2014). Thus, the main parts in MEMORAe CWE are: the semantic map of the organizational ontology, the list of user groups, and the spaces of the selected user groups (see Figure 6.2).

The semantic map is a graphical representation of the organizational ontology. Once an organization is chosen in MEMORAe CWE, users can browse the corresponding semantic map (see Figure 6.2). Specifically, each node, with its own description, symbolizes a concept³ defined in the organizational ontology. When users click on a particular node in the map, resources indexed with this node in user groups become displayable within MEMORAe CWE.

Each user group provides a sharing space for its members to collaborate with each other and interact with resources from different integrated collaborative tools. Specifically, a resource can be accessible and visible in different user groups as wanted by the users. In this process, MEMORAe CWE doesn't change the resource itself in its original storage but its reference and index keys in MCC (see Figure 4.9).

Each user has at least one user group accessible by all members of the chosen organization. The other user groups include at least two members (including the user himself/herself) that have a common goal to achieve within the chosen organization. All available user groups are listed in the left side of MEMORAe CWE (see Figure 6.2).

For understanding how MEMORAe CWE presents and collects the collaboration context through its contextual dimensions and factors⁴, this section first explains its usage based on the scenario presented in Section 1.2. Then, we illustrate an application of the prototype at the University of Technology of Compiègne (UTC).

6.3.1 Usage

MEMORAe CWE enables users to collaborate within user groups⁵. In the scenario, Emma, Lucie, and Marinela are working together in a user group. Meanwhile, Emma also collaborates with her supervisors, Elsa and Marie, on her thesis in another user group. In MEMORAe CWE, these two pertinent user groups can constitute an organization whose members collaborate with each other to achieve relevant goals. Thus, Emma, Lucie, Lisa, Elsa, and Marie are members of an organization *Laboratory X* but belong to two distinct user groups: *Context-aware application project* and *Emma's thesis*. To reach these user groups in MEMORAe CWE, Emma needs to first log in and choose the organization *Laboratory X* (see Figure 6.3).

³Each concept is defined as owl:Thing in Figure 4.3.

⁴The collaboration context contains eight dimensions: *Goal, Collaborator, Activity, Resource, Time, Location, Relation, and Satisfaction*. Each of them consists of several relevant contextual factors. (cf. Section 4.2)

⁵These user groups are defined as instances of mcc:UserGroup in Figure 4.3.

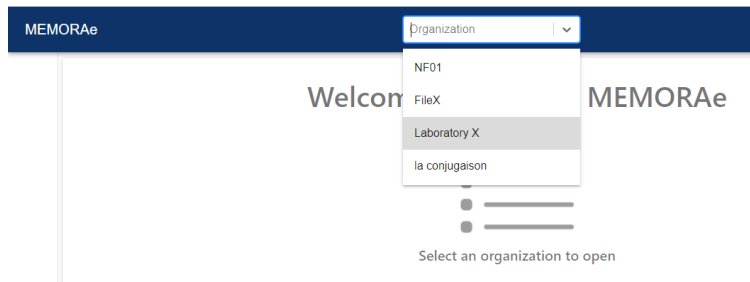


FIGURE 6.3: Selecting an organization in MEMORAe CWE.

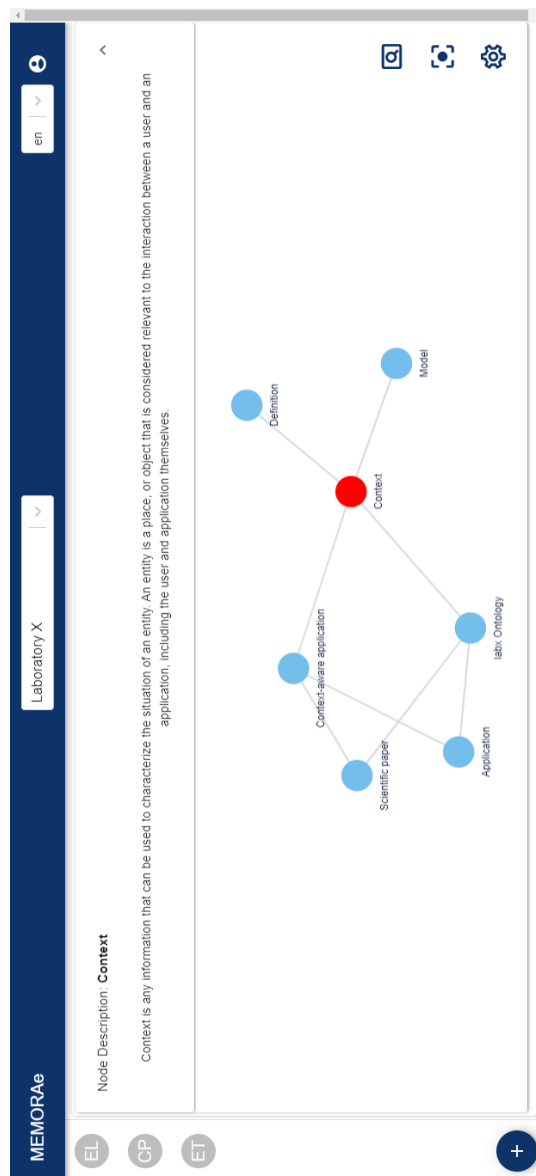


FIGURE 6.4: Organizational knowledge graphs in MEMORAe CWE.

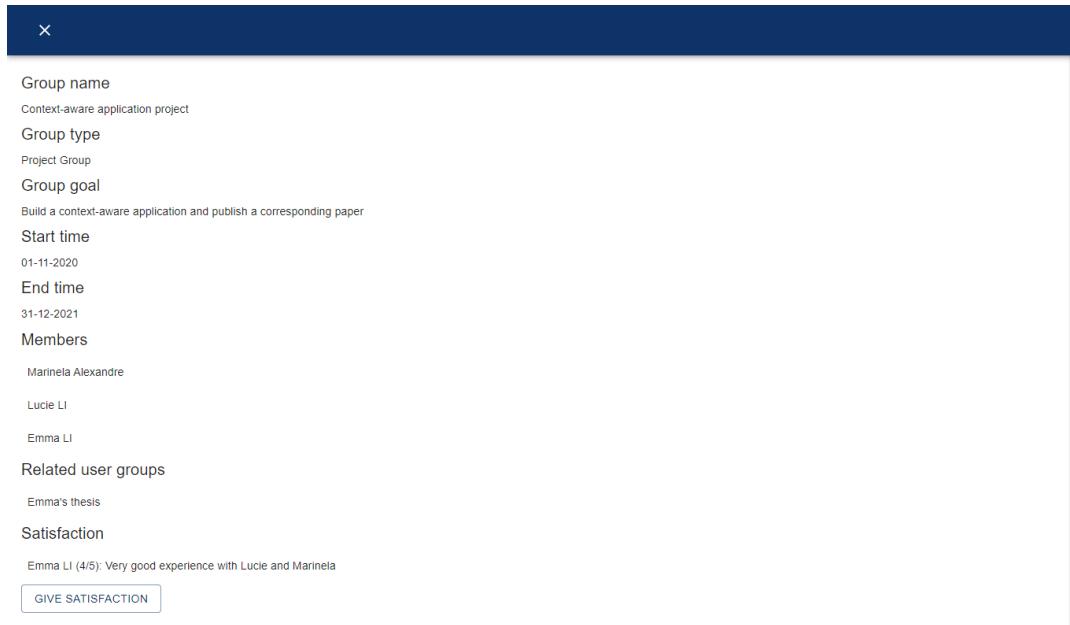


FIGURE 6.5: Group profile interface of the user group *Context-aware application project*.

After choosing the organization, Emma then can navigate its semantic map (see Figure 6.4), which represents a shared vocabulary that describes the goals of collaboration within the chosen organization. Regarding the goals of Emma's two collaborations, certain concepts and their relationships are defined in the corresponding organizational ontology. All members of the chosen organization can view the map (see Figure 6.4) in MEMORAe CWE.

Besides, a detailed description of the goal within each user group can be found in the group profile interface. For example, the goal profile interface of the user group *Context-aware application project* is shown in Figure 6.5. Apart from *Goal*, this interface also contains other contextual dimensions: *Time*, *Collaborators*, *Relation*, and *Satisfaction*. Each dimension consists of certain contextual factors. In Figure 6.5, *Time* includes 'Start time' and 'End time'; *Collaborator* is described by 'Members'; *Relation* is represented by 'Related user groups'; *Satisfaction* is composed of members' satisfaction degrees and their comments.

Meanwhile, users can consult their user profiles in MEMORAe CWE. For instance, Marinela's user profile within MEMORAe CWE is shown in Figure 6.6, indicating her demographic information and abilities (as defined in Figure 4.7).

As for *Resource*, users can progress their collaborations and access diverse resources within user groups of MEMORAe CWE. Each resource is not only accessible and visible in user groups of MEMORAe CWE, but also indexed with a concept of collaboration goals, represented by a node in the semantic map (as defined in Figure 4.9). For instance, the scientific paper added by Emma is accessible in the user group *Context-aware application project* and indexed with the node *Context-aware application* (see Figure 6.7). Notably, resources are visible and accessible only with the indexed nodes. While Emma is selecting

User Information

First Name
Marinela

Last Name
Alexandre

Birthday
01-10-1989

Gender
Female

Affiliation
Sorbonne universités, Université de technologie de Compiègne, CNRS UMR 7253, HEUDIASYC

Education
Computer science Master

Phone Number
06 50 95 05 17

Email work
marinela.alexandre@hds.utc.fr

Language English
Native language

Language French
European level C1

Interest
Web application

[CLOSE](#) [SAVE](#)

FIGURE 6.6: Marinela's user profile interface in MEMORAE CWE.

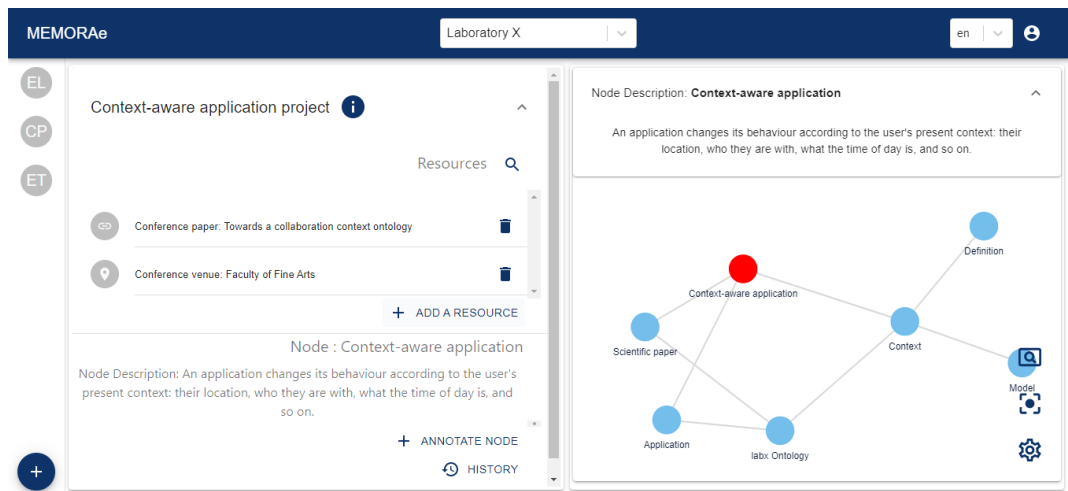


FIGURE 6.7: Emma's interface in MEMORAe CWE.

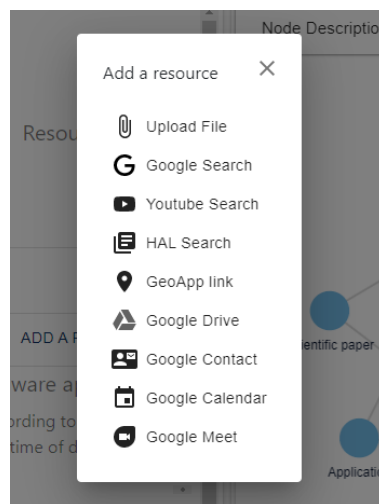


FIGURE 6.8: Collaborative tools integrated into MEMORAe CWE.

another node in the semantic map, the available resources in the user group *Context-aware application project* are different.

In MEMORAe CWE, users can open multiple user groups to view resources simultaneously. All these resources are contained either in MEMORAe CWE, or in an integrated collaborative tools (see Figure 6.8). In Emma's scenario, the cited conference paper is a resource stored in HAL but accessible and visible within user groups of MEMORAe CWE. While Emma was adding this paper in MEMORAe CWE, she first searched for this paper through the API of HAL (see Figure 6.9), then indexed it with the node *Context-aware application* in the user group *Context-aware application project* (see Figure 6.10).

Moreover, users are able to conduct various types of *activities* on resources, including accessing, creating, modifying, sharing, and deleting resource activities (as defined in Figure 4.10). For example, Lucie can create a personal vote and/or annotation on the paper added by Emma in the user group *Context-aware application project*. All these activities are performed

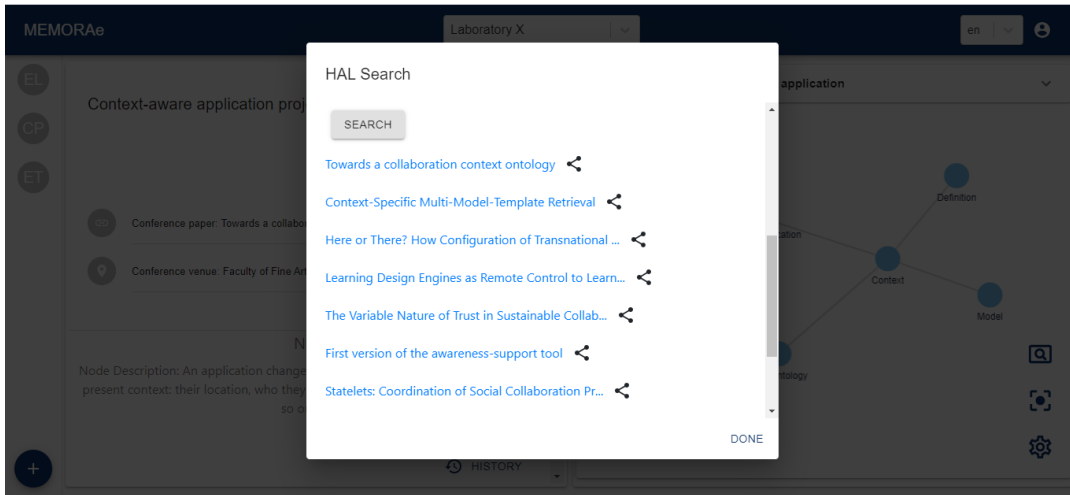


FIGURE 6.9: Searching papers through API of HAL.

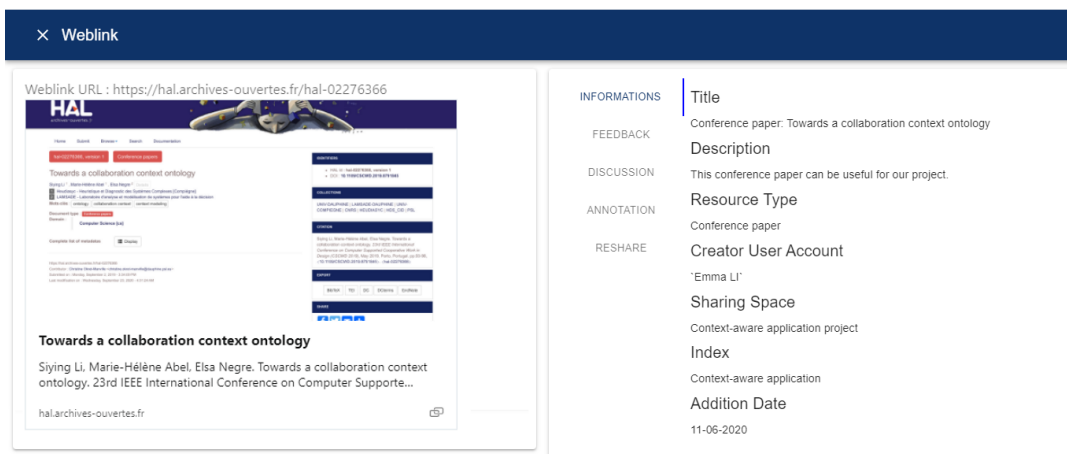


FIGURE 6.10: Detailed information of the conference paper.

My activities in the sharing space

ADD RESOURCE DELETE RESOURCE ACCESS RESOURCE UPDATE RESOURCE SHARE RESOURCE CREATE RESOURCE

Add Resource Activities

Resource Type	Resource Title	Index	Date	Actor
Weblink	Conference paper: Towards a collaboration context ontology	Context-aware application	11-06-2020	Emma LI

Rows per page: 10 1-1 of 1

FIGURE 6.11: Emma’s activities in the user group of *Context-aware application project*.

My activities in the organization

CONCEPT NAVIGATION SYSTEM NAVIGATION ADD RESOURCE DELETE RESOURCE ACCESS RESOURCE UPDATE RESOURCE SHARE RESOURCE CREATE RESO

Concept Navigation Activities

Visited Concept Name	Date
Context	11-06-2020
Context	11-06-2020
Context-aware application	11-06-2020
Context	11-06-2020
Model	11-06-2020
Context-aware application	11-06-2020

Rows per page: 10 1-6 of 6

FIGURE 6.12: Emma’s activities in the organization *Laboratory X*.

within user groups and/or organizations of MEMORAe CWE, thus traced by user groups and organizations. For the activity traces in user groups (see Figure 6.11), they are viewed by activity type, resource type, resource name, index name, date, and actor (user). Meanwhile, users’ activities in organizations have two additional types: navigating collaborative tools and nodes (see Figure 6.12). These traces can be used to assess the progress and status of users’ collaborations within MEMORAe CWE.

Finally, MEMORAe CWE provides users an opportunity to import and view their external activities in other *locations* beyond MEMORAe CWE. For instance, in the website HAL, users’ collaborative activities in scientific research are traced by their publications. Through the API of HAL, users can extract and share such information within MEMORAe CWE. In the scenario, Emma’s activities about the previously published conference paper are shared in the organization *Laboratory X* (see Figure 6.13). Moreover, users can also record useful geographical locations (as defined in Figure 4.3) during their collaborations. For example, while Emma presented the conference paper, Emma applied a stand-alone web application GeoAPP to save geographical information about the conference venue. This allows her to share such information within MEMORAe CWE (see Figure 6.14).

× My activities in the organization DONE

Concept Navigation System Navigation **External collaborations** Add Resource Delete Resource Access Resource Update Resource Share Res >

External collaborations from HAL 🔍 ☰ ☰


Title	Type	Co-author(s)	Publication Date	Key words	Organization of delivery
Towards a collaboration context ontology	Conference paper	Maria-Hélène ABEL, Elsa NEGRE. Keywords	05-2019		Heudiasyc - Heuristique et Diagnostic des Systèmes Complexes;

Rows per page: 10 ▾ 1-1 of 1 < >

FIGURE 6.13: Emma’s external collaborative activities in the organization *Laboratory X*.

× GeoLocation

GeoLocation URL : localhost:3000/api/place/5fa562d80c342b2060067c9b



INFORMATIONS

FEEDBACK

DISCUSSION

ANNOTATION

RESHARE

Title
Conference venue: Faculty of Fine Arts

Description
The conference was presented here to other researchers.

Resource Type
GeoLocation

Longitude
-8.6008

Latitude
41.145941

Creator User Account
"Emma LI"

Sharing Space
Context-aware application project

Index
Context-aware application

Addition Date
11-06-2020

FIGURE 6.14: The location of conference venue in MEMORAE CWE.

6.3.2 Application

This section presents an application of the prototype MEMORAe CWE within an engineering course NF01 at University of Technology of Compiègne (UTC). This application was realized in the spring semester, from 25 February to 12 July 2019. NF01 is an "Algorithms and Programming" course for first year students at UTC. Consequently, 73 students and 3 teachers of this course are users of MEMORAe CWE.

During the course NF01, all students and teachers are members of an organization *NF01* within MEMORAe CWE. In this organization, the teachers in charge set up the organizational ontology (see Figure 6.15) to describe the content and structure of the course NF01. This ontology is presented by means of a semantic map. Each node in the graph represents a notion about the course NF01.

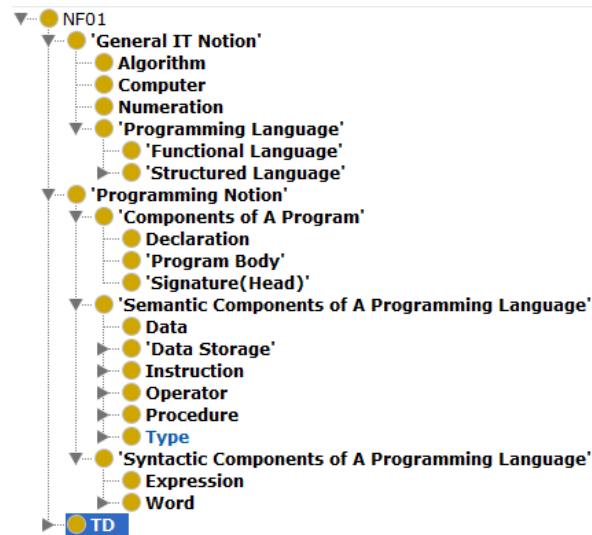


FIGURE 6.15: Partial organizational ontology of the course NF01.

Particularly, members of the organization *NF01* belong to 23 different user groups. These user groups are classified into three types: 1 user group accessible by all members of the organization *NF01*, 5 user groups of practical classes (i.e., TD1, TD2, TD3, TD5, and TD7) comprising students and a teacher in the corresponding practical class, and 17 user groups of projects only consisting of 2, 3 or 4 students and a responsible teacher. These students and teachers can apply 5 integrated collaborative tools, including Google Search (search engine), Google Drive (document sharing and management), YouTube (video searching), File Upload (document storage), and Weblink Share (weblink management).

Students in a user group can then interact with each other to learn the course NF01 collaboratively under the supervision of the teacher. They are also able to submit their project report and/or results to the teacher within user groups. Meanwhile, teachers can share pedagogical documents with students in user groups of practical classes. All these reports, results

and pedagogical documents are accessible resources within user groups, which are also indexed with at least a node in the semantic map. This permits students to link and understand various resources with notions in the course NF01. Moreover, students' and teachers' activities on resources are traced by user groups, such as adding, accessing, and deleting resource activities. This permits students and/or teachers to view other members' activity traces in a user group.

Besides, all students and teachers are provided with a private working space only accessible by themselves, which allows them to index and organize their personal resources within the organization *NF01*.

At the end of the semester, some students gave us their feedback about their experience with the prototype MEMORAe CWE. The complete survey is available in Appendix C. The responses can be summarized as follows:

- 72.7% students felt that MEMORAe CWE could help them revise the course NF01.
- 81.8% students thought MEMORAe CWE could organize resources for them.
- 72.7% students considered MEMORAe CWE as a useful environment to allow them collaborate with others (students and/or teachers).
- Students averagely rated 5.9 (out of 10) for our prototype as a web-based CWE.
- Students averagely rated 5.8 (out of 10) for our prototype as a learning support in the course NF01.
- 72.7% students evaluated that consulting other users' activity traces could help them to identify people that are related to a notion in the course NF01.
- 45.5% students said they would be interested in receiving context-aware collaborator recommendations in MEMORAe CWE.

6.4 Discussion

Regarding the relationship between web-based CWEs and collaborative SoISs, an architecture of an ontology-based collaborative SoIS (Saleh and Abel, 2016) was considered from the viewpoint of web-based CWEs. Based on this architecture, the ontology-based collaboration context model MCC can serve as an ontological knowledge base in web-based CWEs, which can not only process heterogeneous information from different collaborative tools, but also manage it within the context of users' collaborations. From a semantic perspective, this is mainly approached through two parts: T-Box, and A-Box. Specifically, the T-Box enables us to define a uniform semantic structure of the information in a web-based CWE. Such a structure is unconnected to the representation manner or the language of the information,

but is only concerned with its meaning. In such a structure, different levels of information representation can be specified in ontologies, e.g., an organizational ontology for describing the goals of collaboration within an organization. Based on this structure of T-Box, ontology can interpret heterogeneous information from different collaborative tools. Then, the A-Box instantiates the matching information into the corresponding instances. Also, these instances are expressed using ontology, making them interpretable by the tools integrated into the web-based CWE. Thus, the information contained in a web-based CWE can be represented in a unified and standardized way, through which useful knowledge can be reasoned and extracted.

Implementing the ontology-based collaboration context model MCC into web-based CWEs also enhances interoperability between collaborative tool. Particularly, *RESTful API* (see Figure 6.1) provides uniform interfaces to access and manipulate information, allowing them become directly accessible components in web-based CWEs. Each component refers to an instance in the A-Box. This signifies that all instances are available, comprehensible and operable in web-based CWEs. In other words, organizing heterogeneous information into instances simplifies information communications between different tools, thus improving information access and management in web-based CWEs.

To summarize, through considering the architecture of the collaborative SoIS from the viewpoint of web-based CWEs, the ontology-based collaboration context model MCC can be implemented into web-based CWEs. This answers a main question of this thesis: how to integrate the collaboration context model into web-based CWEs? Besides, with MCC, users in web-based CWEs can organize heterogeneous information within the context of their collaborations.

Also, this work has some limitations. Due to the limited time, we are unable to add functionalities related to context-aware collaborator recommendations in the prototype, *MEMORAe* CWE. It would be a good practice to develop such functionalities and let users evaluate them during their collaborations.

6.5 Chapter summary

In this chapter, our work is mainly about the third question listed in Section 1.1.3: how to implement and integrate the collaboration context model into web-based CWEs?

Firstly, an architecture of an ontology-based collaborative SoIS (Saleh and Abel, 2016) is considered from the viewpoint of web-based CWEs, given the relationship between web-based CWEs and collaborative SoISs. Thanks to this architecture, the ontology-based collaboration context model MCC is implemented in web-based CWEs as an ontological knowledge base. Then, a corresponding web-based CWE prototype, *MEMORAe* CWE, is developed. We explored its main functionalities and explained how the collaboration context can

be presented and collected within it. Besides, the prototype *MEMORAe CWE* was tested by the students and teachers of a course NF01 at UTC. Their feedback were also shown. Lastly, we discussed our work with *MEMORAe CWE* and how it responds to the main questions of this thesis.

The next chapter will focus on the last question of this thesis: how well do our context-aware collaborator recommendations algorithms perform in terms of accuracy and time efficiency?

Chapter 7

Experiments

7.1 Introduction

With the ontology-based model MCC (cf. Chapter 4) and the context-aware recommendation algorithms (cf. Chapter 5), we can process the collaboration context by means of an ontology-based semantic similarity and thus produce context-aware collaborator recommendations for users. Then the last question is: are our context-aware collaborator recommendations algorithms relevant in terms of accuracy and time efficiency? To solve it, we utilize a public dataset of scientific collaborations, which is extracted from multiple academic bibliographies: DBLP¹, Microsoft Academic Graph (MAG)², and AMiner³. Particularly, we apply all the context-aware recommendation algorithms (i.e., PreF1, PoF1, and PoF2) in this dataset and compare their performances in terms of accuracy and time efficiency. These results allow us to analyze both advantages and disadvantages of employing the ontology-based semantic similarity in producing context-aware collaborator recommendations.

Therefore, the rest of this chapter is organized as follows. Section 7.2 introduces the dataset we use and how its scientific collaborations can be presented by the ontology-based collaboration context model. Section 7.3 then shows and evaluates the results obtained from this dataset. Next, we discuss how well these results perform in terms of accuracy and time efficiency. A summary can be found at the end of this chapter.

7.2 Dataset

To test and evaluate the performances of the context-aware collaborator recommendation algorithms, we aim to search for datasets where data is retrieved from real users and their collaborations in web-based CWEs. Without finding such datasets, our alternative is to consider an academic article as a scientific collaboration and thus to make use of academic publication datasets for the experiments. While constructing such datasets, MCC permits us to gather the

¹<https://dblp.org/>

²<https://academic.microsoft.com/home>

³<https://www.aminer.org/>

collaboration context from different sources while calculating the ontology-based semantic similarities (cf. Chapter 5). In other words, the dataset in our experiments can consist of heterogeneous information from multiple academic bibliographies. These bibliographies refer to different collaborative tools integrated in a web-based CWE.

Therefore, we choose to utilize a dataset (Tang et al., 2008)⁴ designed for research purpose only. It includes academic articles and their citation relationships until 2019-05-05, which are extracted from DBLP, MAG, and AMiner. Each academic article in this dataset can be described by different fields (see Table 7.1). Specifically, not all academic articles contain values of all the fields shown in Table 7.1. Some values can be missing. For example, articles that are not included in MAG don't have values for the fields *fos.name* and *fos.w*. Because values of these two fields are obtained from MAG.

Field Name	Field Type	Description
id	string	paper ID
title	string	paper title
authors.name	string	author name
author.org	string	author affiliation
author.id	string	author ID
venue.id	string	paper venue ID
venue.raw	string	paper venue name
year	int	published year
fos.name	string	paper fields of study
fos.w	float	fields of study weight
references	list of strings	paper references ID
n_citation	int	citation number
page_start	string	page start
page_end	string	page end
doc_type	string	paper type:journal, book, etc.
publisher	string	publisher
volume	string	volume
issue	string	issue
issn	string	issn
isbn	string	isbn
doi	string	doi
pdf	string	pdf URL
url	list	external links
indexed_abstract	dict	indexed abstract

TABLE 7.1: Fields of academic articles in the dataset.

⁴This dataset can be downloaded from <https://www.aminer.org/citation>.

This dataset contains 4,107,340 articles and 36,624,464 citation relationships. Most of these articles are tagged with research domains. These research domains are organized in a non-mutually exclusive hierarchy with 19 top-level domains: Art, Biology, Business, Chemistry, Computer science, Economics, Engineering, Environmental science, Geography, Geology, History, Materials science, Mathematics, Medicine, Philosophy, Physics, Political science, Psychology, and Sociology. This allows us to separate the dataset into 20 blocks. Each block contain articles in one top-level domains. Especially, the 20th block includes the articles that are not tagged with any top-level domains. The article numbers in each block⁵ are summarized in Table 7.2.

Block	Article number	Block	Article number
Art	4944	History	3163
Biology	36941	Materials science	8907
Business	41406	Mathematics	904745
Chemistry	13808	Medicine	32795
Computer science	2606722	Philosophy	5847
Economics	37985	Physics	25715
Engineering	248640	Political science	9064
Environmental science	3160	Psychology	49972
Geography	14450	Sociology	15663
Geology	8614	Others	35518

TABLE 7.2: Article numbers in each block.

Particularly, each article in the dataset represents the fact that its authors have collaborated once in certain research domains. During such a collaboration, the authors work together to write the corresponding article. This implies that an academic article is the result of a scientific collaboration. This permits us to consider an article as a scientific collaboration. Accordingly, the side information of an article (shown in Table 7.1) belongs to the context of a scientific collaboration.

MCC defines eight contextual dimensions of collaboration and their associated factors (cf. Section 4.2). In scientific collaborations, all these dimensions and factors can be represented as follows (see Table 7.3). Each collaboration is identified by *id* and *title* of an article. For dimension Relation, we filter out irrelevant values of *references* using the list of the same block’s articles’ IDs. Because this dimension indicates relation between collaborations in a same research domain. We do not consider relations between collaborations in different fields. Besides, dimension Activity includes only one type of collaborators’ actions: writing academic paper activity, but with their different contributions. As for dimension Satisfaction, we do not have any information or relevant information that can replace this dimension.

⁵The sum of article numbers in all the 20 blocks is 4,108,059, more than the article number (4,107,340) in the dataset. 709 articles are part of more than one block, while the other 4,106,632 articles only belong to one block. Specifically, among the 709 articles, 698 articles are in two blocks; 9 articles belong to three blocks; 1 article is even in four blocks.

Contextual dimension	Contextual factor	Field Name (in the dataset)
Goal	Focused field of study	fos.name
Collaborator	Name of actor	authors.name
	ID of collaborator	authors.id
Activity	Name of actors	authors.name
	ID of actors	authors.id
	Activity type	"Writing academic article activity"
	Contribution of actors	Authors' order in the author list
Resource	ID of cited articles	references
Time	End time	year
Location	Location involved	authors.org
Relation	ID of related collaborations (in a same block)	references
Satisfaction		

TABLE 7.3: Transformation from articles' side information into the contexts of scientific collaborations.

An example of a scientific collaboration with its context is shown below.

```

1 {"id": "1023950486",
2  "title": "Structuring and reusing knowledge from historical
   events for supporting nuclear emergency and remediation
   management",
3  "Collaborator": [
4    {"name": "Stella Moehrle", "id": "2185715410"},
5    {"name": "Wolfgang Raskob", "id": "1699906302"}
6  ],
7  "Location": [
8    "Institute for Nuclear and Energy Technologies, Karlsruhe
   Institute of Technology (KIT), Hermann-von-Helmholtz-
   Platz 1, 76344 Eggenstein-Leopoldshafen, Germany"
9  ],
10 "Activity": [
11  {"name": "Stella Moehrle", "id": "2185715410", "type": "
   Writing academic article activity", "contribution": 1},
12  {"name": "Wolfgang Raskob", "id": "1699906302", "type": "
   Writing academic article activity", "contribution": 2}
13 ],
14 "Goal": [

```



```

15   "Computer science", "Reuse",
16   "Management science", "Emergency management",
17   "Decision support system",
18   "Case-based reasoning", "Structuring", "History"
19 ],
20 "Time": 2015,
21 "Relation": [],
22 "Resource": [
23     "2032330087",
24     "2041661842",
25     "2087780521",
26     "2126385963"
27 ]
28 }

```

With such a transformation, each article is a scientific collaboration with its own context. Accordingly, each block of articles can be considered as a set of scientific collaborations in the corresponding top-level research domain. Specifically, every set is composed of 1000 articles selected randomly from a block due to the different article numbers in these blocks (see Table 7.3). These articles are arbitrarily divided into two parts, representing separately training collaboration and testing collaborations. Particularly in the context-aware collaborator recommendation algorithms (i.e., PreF1, PoF1, and PoF2 in Algorithm 1), training collaborations constitute the set X , providing known information (e.g., known ratings, known contexts of scientific collaborations). Meanwhile, each testing collaboration can be c . For every author u in a testing collaboration c , context-aware collaborator recommendations are generated and tested in our experiments⁶.

Particularly, for all scientific collaborations, we have $T = 7$ and $T_{quan} = 6$ (i.e., Goal, Collaborator, Activity, Resource, Location, and Relation). Only one dimension Time is described by quantitative objects (i.e., $T_{qual} = 1$). Therefore, based on Equation 5.13 and 5.15, the ontology-based semantic similarity between two scientific collaborations (c and d)⁷ is:

$$\begin{aligned}
 S(d, c) &= S_1(d, c) + S_2(d, c) \\
 &= \sum_{t=1}^6 \sum_{g_1=1}^{|P^{dc,t}|} \frac{|O_{p_{g_1}^t}^d \cap O_{p_{g_1}^t}^c|}{|O_{p_{g_1}^t}^d \cap O_{p_{g_1}^t}^c| + \alpha |O_{p_{g_1}^t}^d - O_{p_{g_1}^t}^c| + \beta |O_{p_{g_1}^t}^c - O_{p_{g_1}^t}^d|} \times \frac{IC(P^{d,t})}{\sum_{h=1}^7 IC(P^{d,h})} \\
 &\quad + \sum_{g_2=1}^{|P^{dc,Time}|} \frac{1}{\gamma |o_{p_{g_2}^{Time}}^d - o_{p_{g_2}^{Time}}^c| + 1} \times \frac{IC(P^{d,Time})}{\sum_{h=1}^7 IC(P^{d,h})}
 \end{aligned} \tag{7.1}$$

⁶Here, u and c are two variables in the formulated problem of context-aware collaborator recommendations (cf. Section 5.5).

⁷Here, c is a testing collaboration; $|X|$ represents the number of training collaborations; $d(d \in X, d \neq c)$ is a training collaboration.

where $|P^{dc,t}|$ is the number of predicate types belonging to the t^{th} contextual dimension that both collaborations c and d relate; $|P^{dc,Time}|$ is the number of predicate types belonging to the contextual dimension Time that both collaborations c and d relate; $\alpha = \beta = \gamma = 1$. The range of $S(d, c)$ is $[0, 1]$.

Within this semantic similarity $S(d, c)$, this dataset can be applied to test PreF1, PoF1, and PoF2 algorithms (cf. Section 5.5). This dataset well fits our needs in the experiments for the following reasons:

- In this dataset, authors of an academic article are members of a scientific collaboration. The detailed information of an article (shown in Table 7.1) belongs to the context of this collaboration, which can be represented by the contextual dimensions and their associated factors (shown in Table 7.3). This corresponds to our ontology-based collaboration context model MCC;
- This dataset is constructed by heterogeneous information from different academic bibliographies, corresponding to MCC that the collaboration context comes from different collaborative tools within a web-based CWE;
- From the aspect of volume, this dataset is able to measure the performance of the context-aware collaborator recommendation algorithms (i.e., PreF1, PoF1, and PoF2). These algorithms apply the ontology-based semantic similarity (see Equation 5.18 and 5.15) in different phases to generate recommendations.
- In this dataset, we are able to employ other context-aware recommendation algorithms following Contextual Modeling (CM) method (Adomavicius and Tuzhilin, 2011), such as Context-Aware Matrix Factorization (CAMF)⁸ (Baltrunas, Ludwig, and Ricci, 2011). This allows us to compare context-aware recommendation algorithms following PreF and PoF methods with those of CM method.

Thus, it is reasonable to conduct experiments on this dataset. The results of such experiments can help us analyze whether our solution is relevant in terms of accuracy and time efficiency.

7.3 Experiments

7.3.1 Evaluation methods

Before applying the algorithms on the dataset, we need to be clear about the evaluation methods. To compare the generated context-aware collaborator recommendations with

⁸There are three types CAMF: CAMF-C, CAMF-CI, and CAMF-CC (Baltrunas, Ludwig, and Ricci, 2011). Based on the dataset, CAMF-C can be realized as a baseline algorithm in our experiments.

TABLE 7.4: Classification of collaborators in generate recommendations for an author u in a testing collaboration c .

		Recommended collaborators to u in c ?	
		No	Yes
Real collaborators of u in c ?	No	TN	FP
	Yes	FN	TP

real collaborators in a testing collaboration, we employ three metrics in our experiments: F1, Mean Absolute Error (MAE), and execution time. This enables us to evaluate the performances of the context-aware collaborator recommendation algorithms from two aspects: accuracy and time efficiency.

For evaluation accuracy, we first apply F1 metric in our experiments, which is calculated by the following equation (Van Rijsbergen, 1979; Goutte and Gaussier, 2005):

$$F1 = \frac{TP}{TP + \frac{1}{2}(FN + FP)} \quad (7.2)$$

where TP , FN , and FP are numbers of classified collaborators based on Table 7.4.⁹ The range of F1 is $[0, 1]$. A higher value of F1 indicates more accurate recommendations.

Besides, the MAE metric is also utilized in our experiments, which is popular in evaluating accuracy (Willmott and Matsuura, 2005). This metric signifies the difference between an author u 's actual rating R_{ui} and predicted rating \hat{R}_{ui} for a collaborator i (see Equation 7.3). The lower the value of MAE, the smaller the difference between R_{ui} and \hat{R}_{ui} .

$$MAE = \frac{\sum_{i=1}^m |R_{ui} - \hat{R}_{ui}|}{m} \quad (7.3)$$

where m is the number of users that can be recommended to u in c .

In addition, time efficiency is equally critical in assessing the context-aware collaborator algorithms (Miller et al., 2004). Therefore, we measure the execution time¹⁰ that each algorithm takes to generate context-aware collaborator recommendations for an author u in a testing collaboration c .

With the three metrics (i.e., F1, MAE, and execution time), our experiments are conducted on a computer with the following properties:

Operating System: Windows 10 Entreprise 64-bit

Processor: Inter(R) Core(TM) i7-8650U CPU @ 1.90GHz

Installed Memory (RAM): 32.00 GB

⁹Particularly, we have $FP + TP = K$ and $FN + TP =$ Numbers of u 's real collaborators in c .

¹⁰In our experiments, execution time is counted in milliseconds.

Software: PyCharm 2019.1 (Professional Edition) + Python 3.7.3

7.3.2 Experiment with different percentages of training collaborations

In this experiment, we change the numbers of training collaborations in the context-aware collaborator recommendation algorithms: PreF1, PoF1, and PoF2 (see Algorithm 1). This is to analyze whether and how the volume of training collaborations influences the generation of context-aware collaborator recommendations. Each experiment uses a different percentage of training collaborations in a set of scientific collaborations¹¹. While running these experiments, F1, MAE and execution time are measured.

Simultaneously, experiments are also performed in another context-aware recommendation algorithm following CM method, CAMF (Baltrunas, Ludwig, and Ricci, 2011), and a 2D recommendation algorithm, PMF (Mnih and Salakhutdinov, 2008). This allows us to compare the performances of PreF1, PoF1, and PoF2 with other algorithms. All results are shown in Figures 7.1, 7.2, and 7.3.¹²

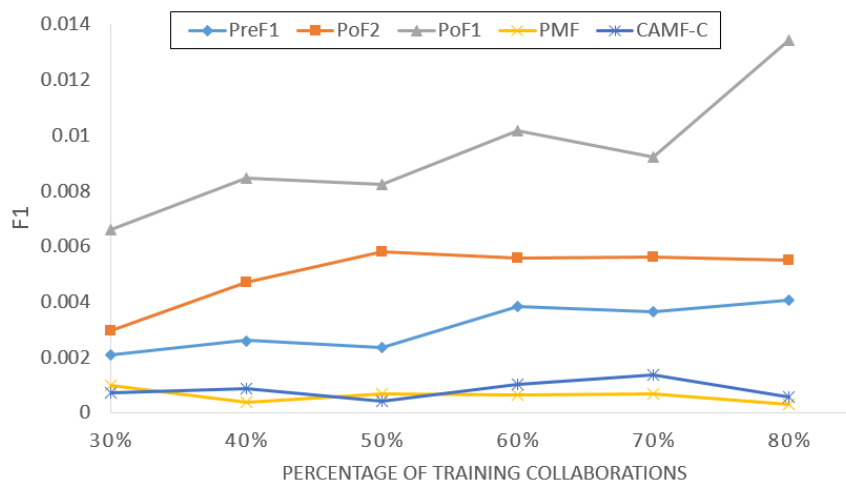


FIGURE 7.1: F1 with different percentages of training collaborations in a set of scientific collaborations.

To conclude, PoF1 obviously outperforms the other algorithms in terms of F1 metric. But it also has the highest values of MAE due to the calculations of the adjusted ratings (cf. Section 5.5). Besides, PreF1 and PoF2 have better performances than PMF and CAMF in both F1 and MAE metrics. This indicates that using the ontology-based semantic similarity leads to more accurate recommendations generated by PreF1, PoF1, and PoF2. As the percentage of training collaborations increases, values of F1 and MAE increases in almost all algorithms (except values of MAE in CAMF algorithm). This implies a common feature of these algorithms: their accuracy rely on the number of training data.

¹¹Here, a corresponding percentage of testing collaborations is $1 - \text{percentage}(\text{training})$.

¹²The algorithms (i.e., PreF1, PoF1, and PoF2) have different values of z (see Algorithm 1). In Figures 7.1, 7.2, and 7.3, PreF1 owns $z = 0$; PoF1 has $z = 1$; PoF2 refers to $z = 2$.

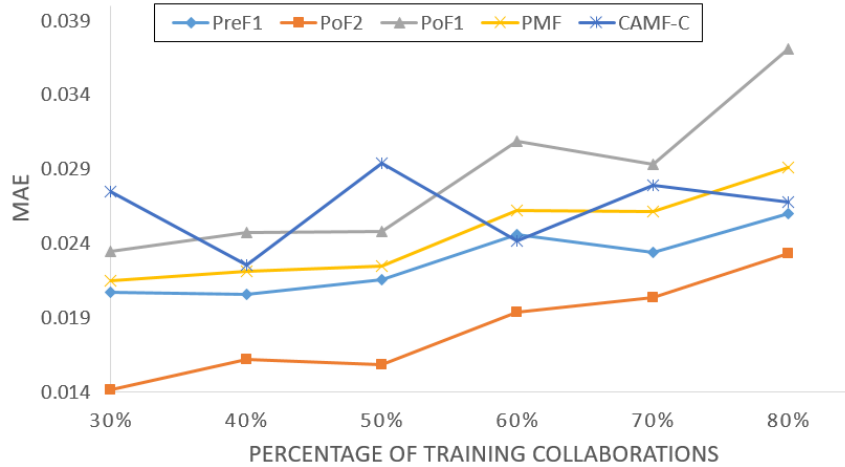


FIGURE 7.2: MAE with different percentages of training collaborations in a set of scientific collaborations.

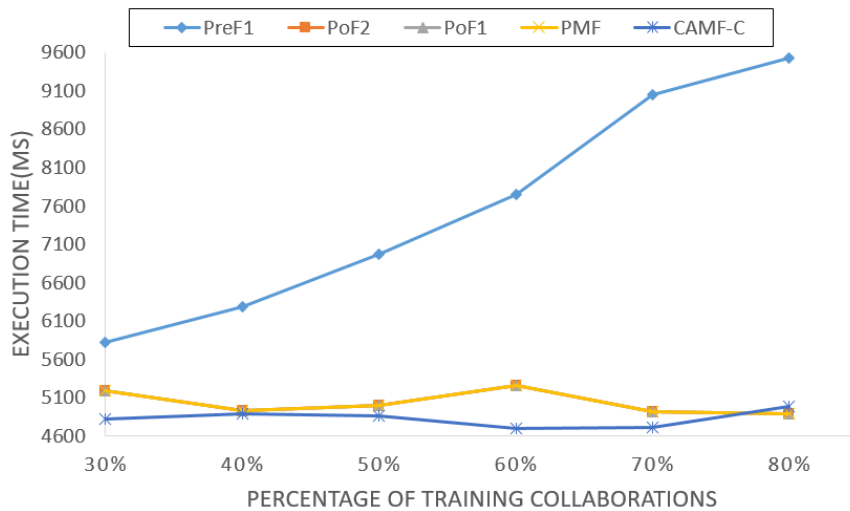


FIGURE 7.3: Execution time with different percentages of training collaborations in a set of scientific collaborations.

As for execution time, PoF1, PoF2, and PMF have almost the same results. And their execution time is very close to that of CAMF because of the short time needed to compute ontology-based semantic similarities. In addition, PreF1 needs the longest execution time compared to other algorithms. The difference between PreF1 and other algorithms is slight, when the percentage of training collaborations is small. But it rises rapidly as the percentage grows, due to the differences in time complexity among these algorithms. Specifically, the time complexity of PreF1 is $O(N^3)$, while that of the others is $O(N^2)$.

7.3.3 Experiment with different values of K

In this experiment, we change the number of generated context-aware collaborator recommendations (i.e., values of K) in the algorithms: PreF1, PoF1, and PoF2 (see Algorithm

1). This is to analyze whether and how the numbers of generated recommendations influences the performances of the three algorithms. Each experiment uses a different K in a set of scientific collaborations. F1, MAE and execution time are computed during these experiments. Similarly, we also perform the experiments in CAMF (Baltrunas, Ludwig, and Ricci, 2011) and PMF (Mnih and Salakhutdinov, 2008). This is to compare the accuracy and time efficiency between these algorithms with different values of K . All results are shown in Figures 7.4, 7.5, and 7.6.¹³

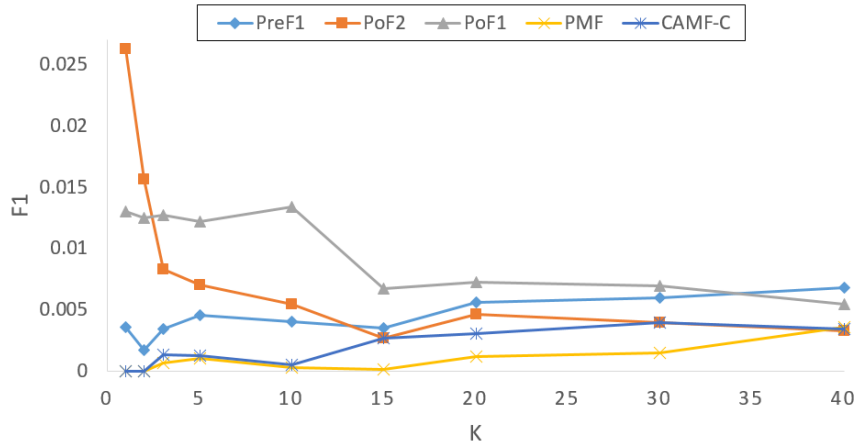


FIGURE 7.4: F1 with different K in a set of scientific collaborations.

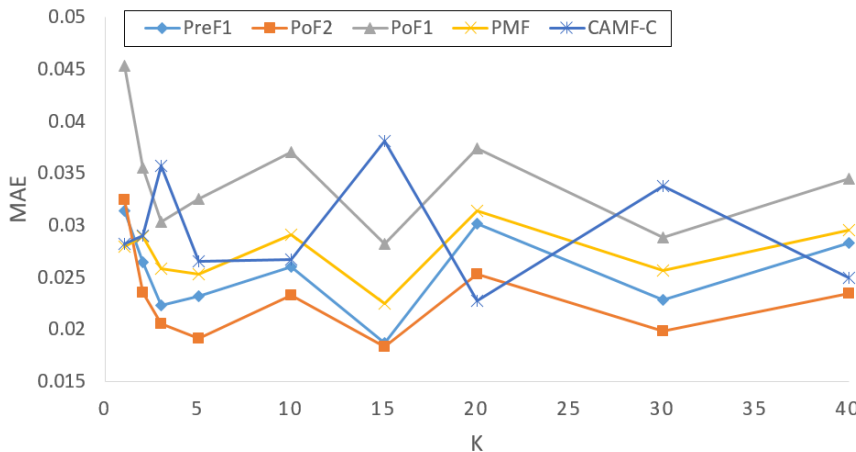


FIGURE 7.5: MAE with different K in a set of scientific collaborations.

In conclusion, the highest values of F1 are achieved by PoF2, PoF1, and PreF1 with different K . Specifically, when K is small, PoF2 has the best performance; when K takes middle values, PoF1 is a better choice; when K is larger than 35, choosing PreF1 seems perfect. This implies that PreF1, PoF1, and PoF2 can generate more accurate recommendations than CAMF and PMF. This also indicates that applying the ontology-based semantic similarity in different phases of the recommendation algorithms (cf. Section 5.1) impacts F1 and therefore leads to varying accuracy of these algorithms. In addition, when K is small, the difference

¹³In Figures 7.4, 7.5, and 7.6, PreF1 owns $z = 0$; PoF1 has $z = 1$; PoF2 refers to $z = 2$ (see Algorithm 1).

of F1 among these algorithms is significant. As K grows, the algorithms tend to converge. This suggests that the ontology-based semantic similarity can augment the accuracy of these algorithms. But its improvements progressively diminish as K grows.

Besides, PreF1 and PoF2 generally have better performances than PoF1, PMF, and CAMF in MAE metric. Similarly, because of the calculations of the adjusted ratings (cf. Section 5.5), PoF1 still has the highest MAE. However, these algorithms show no obvious trend in MAE when K rises. Plus, the values of MAE are extremely unstable in these algorithms. This proves that the MAE acts independently with K but relates to the random division between training and testing collaborations.

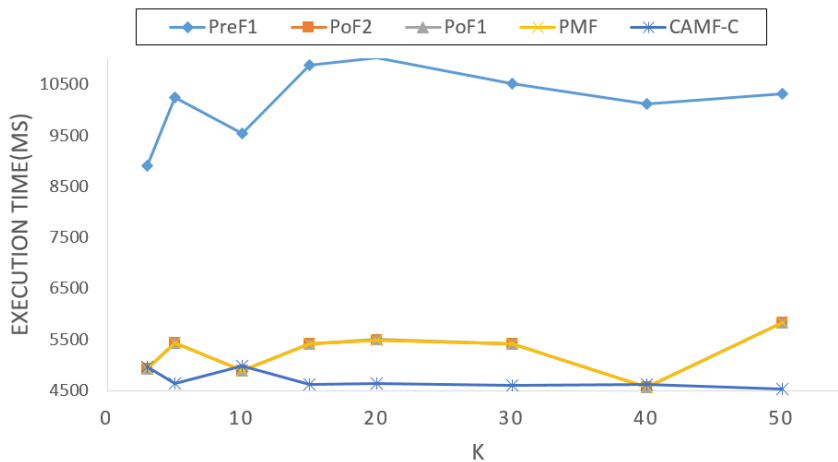


FIGURE 7.6: Execution time with different K in a set of scientific collaborations.

As for execution time, PoF1, PoF2, and PMF still have almost the same results, which are close to that of CAMF. Similarly, PreF1 requires the longest execution time due to its higher time complexity. However, the execution time of all algorithms remain stable as K changes. This indicates that the time efficiencies of these algorithms are also independent with K .

7.3.4 Experiment with different sets of scientific collaborations

In this experiment, we change the set of scientific collaborations in PreF1, PoF1, and PoF2 (see Algorithm 1). Each set contains the same number of scientific collaborations but a distinct number of authors, thus representing a different degree of closeness between these collaborations. This enables us to test whether the improvements brought by the ontology-based semantic similarity work with collaborations having distinct degrees of closeness. Particularly, each experiment uses a different set of scientific collaborations with its own training and testing collaborations. During these experiments, F1, MAE and execution time are recorded. We also perform the experiments in CAMF (Baltrunas, Ludwig, and Ricci, 2011)

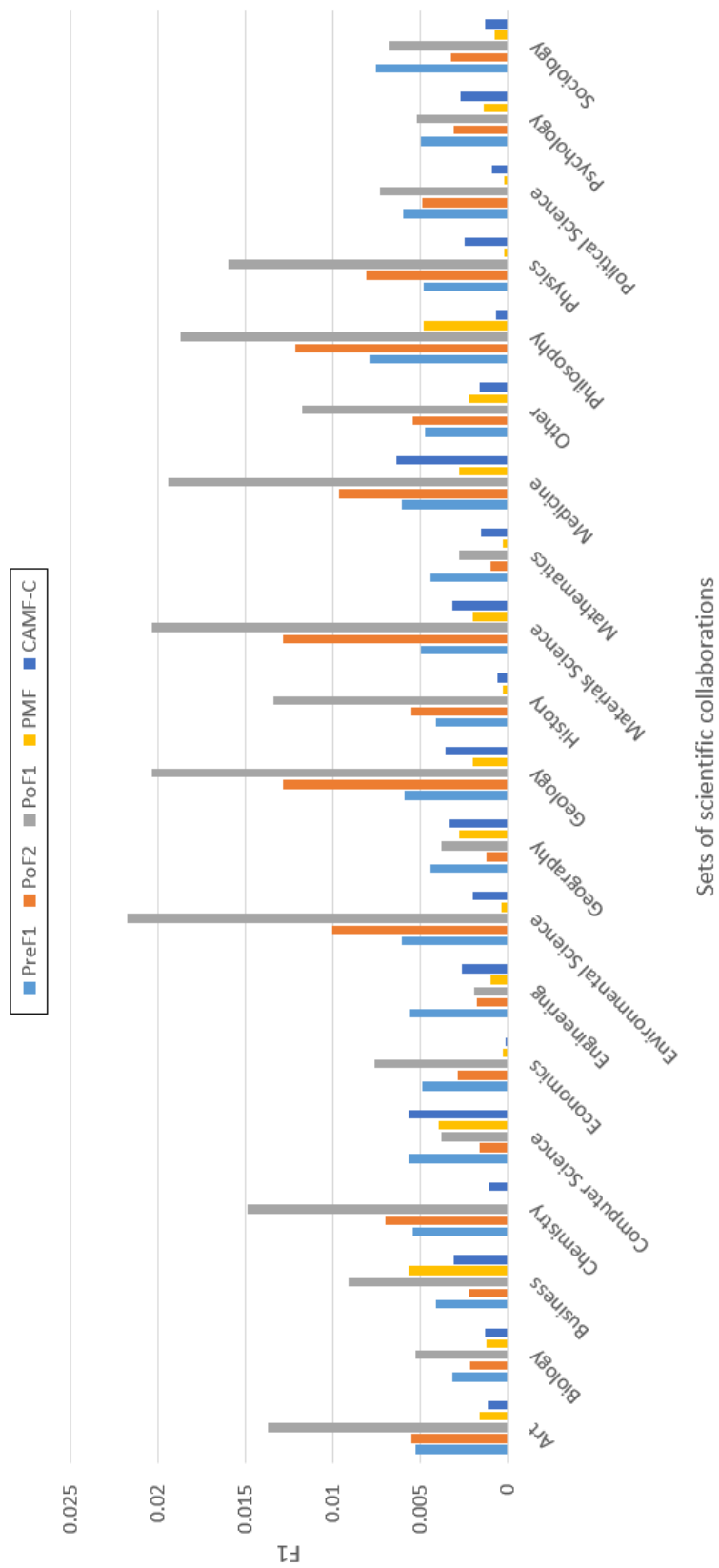


FIGURE 7.7: F1 with different sets of scientific collaborations.

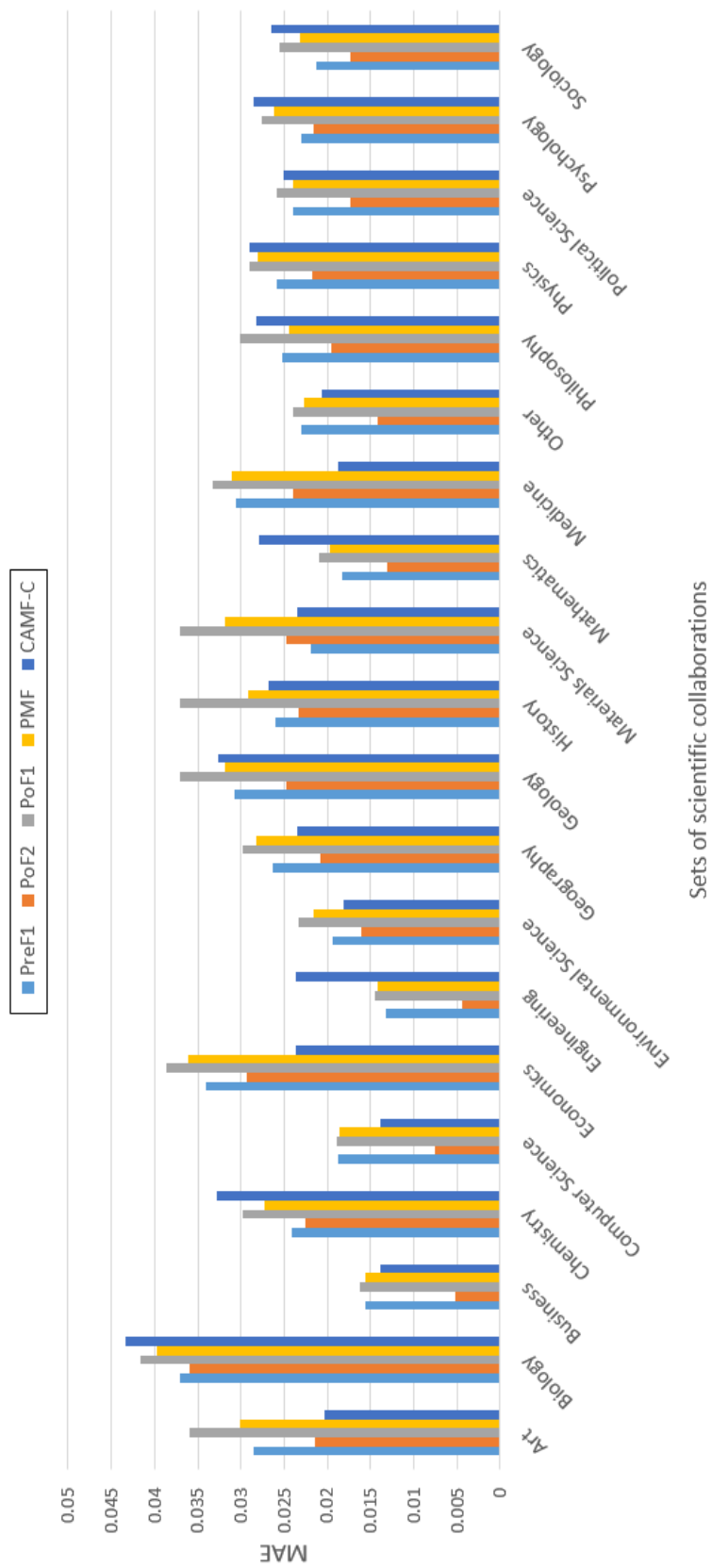


FIGURE 7.8: MAE with different sets of scientific collaborations.

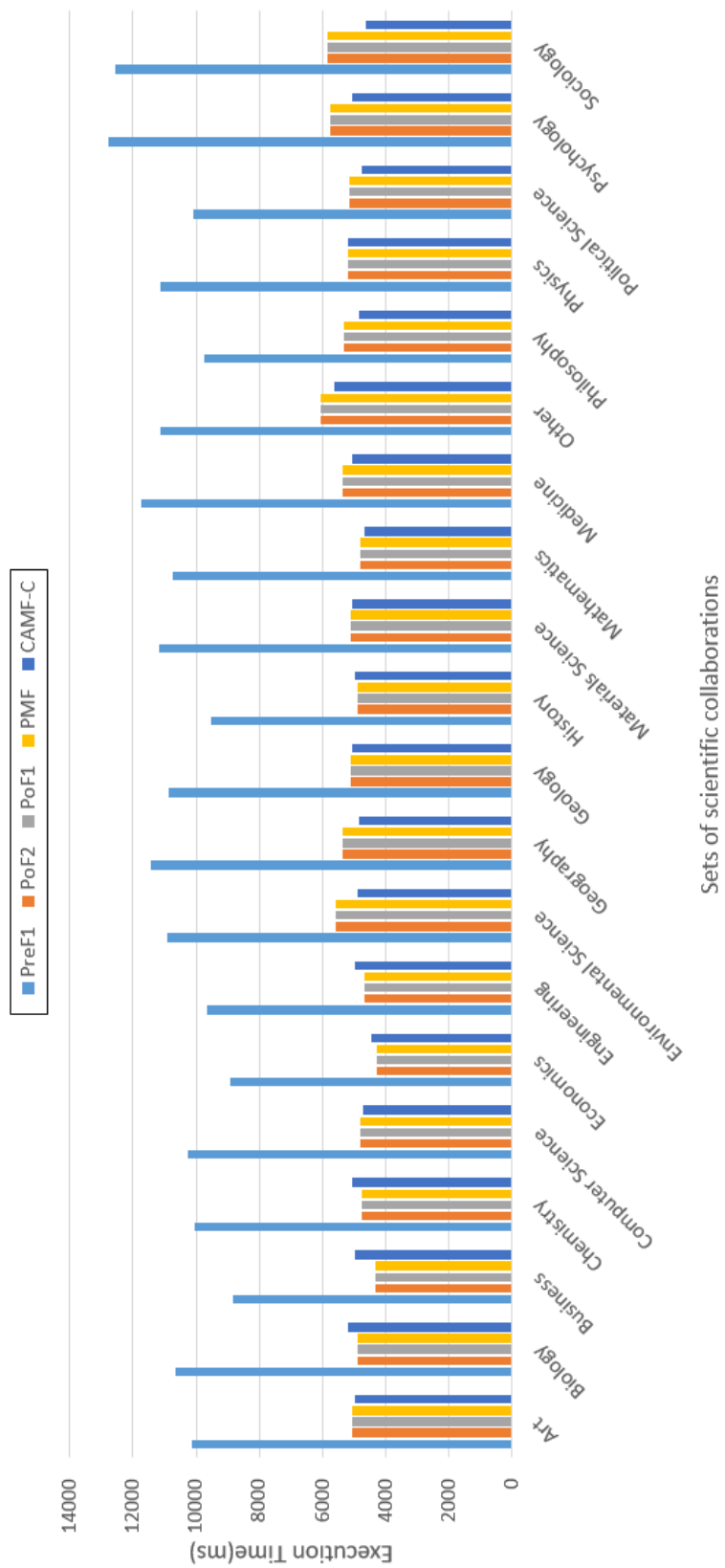


FIGURE 7.9: Execution time with different sets of scientific collaborations.

and PMF (Mnih and Salakhutdinov, 2008). All results are shown in Figures 7.7, 7.8, and 7.9.¹⁴

Overall, PoF2, PoF1, and PreF1 have higher values of F1 than CAMF and PMF in almost all sets of scientific collaborations. This indicates that applying the ontology-based semantic similarity in the context-aware collaborator’s recommendation generation processes indeed improves F1 of the generated recommendations. Specifically, PoF1 clearly exceeds both PoF2 and PreF1.

However, in terms of MAE, its highest values are obtained in PoF1, due to the calculations of the adjusted ratings (cf. Section 5.5). Besides, PoF2 has the best performance of MAE in nearly all these sets, resulting in more accurate recommendations. As for PreF1, it has lower MAE than CAMF and PMF in half of these sets, but higher in the other half. Thus, among these algorithms, PoF2 is a better choice in terms of MAE metric.

Finally, regarding execution time, PoF1, PoF2, PMF, and CAMF have almost the same results. And, PreF1 still takes longer execution time than the others because of its higher time complexity. But no obvious trend in execution time is shown. This implies that the time efficiencies of these algorithms are unrelated to the degree of closeness between collaborations in a set.

7.3.5 Experiment with different semantic similarities

In this experiment, we change the semantic similarity applied in the context-aware collaborator recommendation algorithms: PreF1, PoF1, and PoF2 (see Algorithm 1). This provides us an opportunity to investigate how much our ontology-based semantic similarity (cf. Section 5.4) contributes to generating context-aware collaborator recommendations. Particularly, our ontology-based semantic similarity is inspired and developed from two types of semantic similarities: feature-based and information context-based. To compare our ontology-based semantic similarity with others, we need to utilize existing semantic similarities of these two types in PreF1, PoF1, and PoF2.

Specifically, our experiments involves 5 existing semantic similarities, including Jaccard, Dice, Tversky, TF-IDF, and IC¹⁵ (Jaccard, 1901; Dice, 1945; Tversky, 1977; Batet, Sánchez, and Valls, 2011; Zhang et al., 2008; Sánchez et al., 2012; Rajaraman and Ullman, 2011). Among them, the first three are feature-based similarities, while the rest two belong to information content-based similarities. Each experiment uses a different semantic similarity in a set of scientific collaborations. To analyze their performances in the algorithms, F1, MAE and execution time are employed. All the results are shown in Figures 7.10, 7.11, and 7.12.¹⁶

¹⁴In Figures 7.7, 7.8, and 7.9, PreF1 owns $z = 0$; PoF1 has $z = 1$; PoF2 refers to $z = 2$ (see Algorithm 1).

¹⁵Here, IC represents $IC(c) = -\log p(c)$, where $p(c)$ is the probability of c ’s appearance in an ontology (Batet, Sánchez, and Valls, 2011; Zhang et al., 2008; Sánchez et al., 2012).

¹⁶In Figures 7.10, 7.11, and 7.12, PreF1 owns $z = 0$; PoF1 has $z = 1$; PoF2 refers to $z = 2$ (see Algorithm 1).

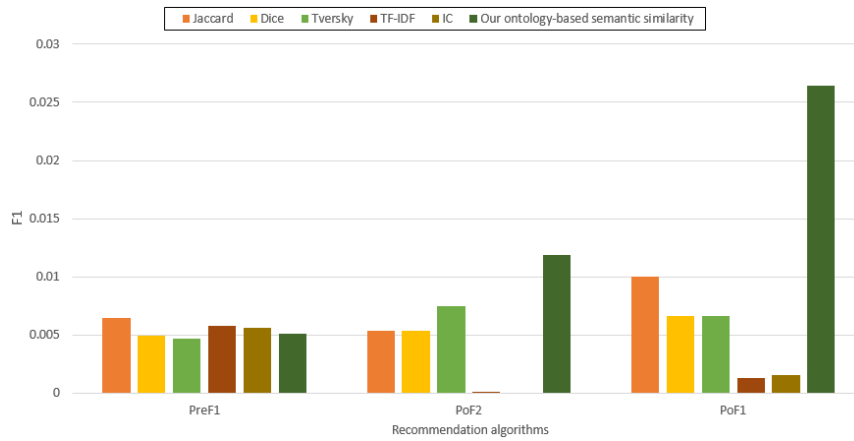


FIGURE 7.10: F1 with different semantic similarities in a set of scientific collaborations.

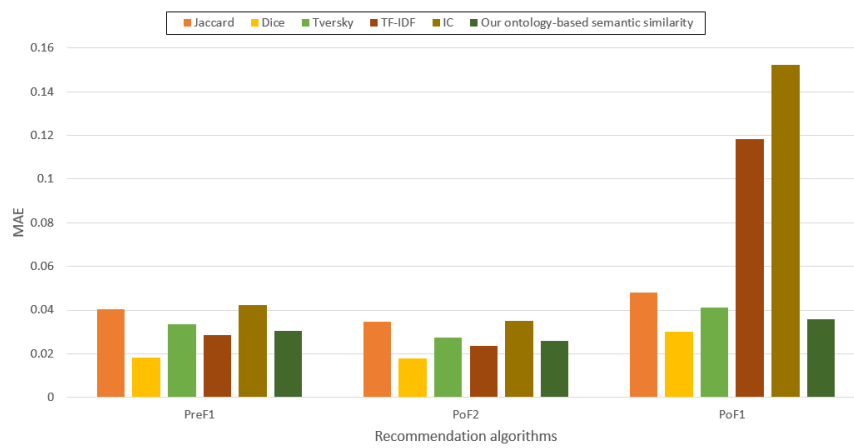


FIGURE 7.11: MAE with different semantic similarities in a set of scientific collaborations.

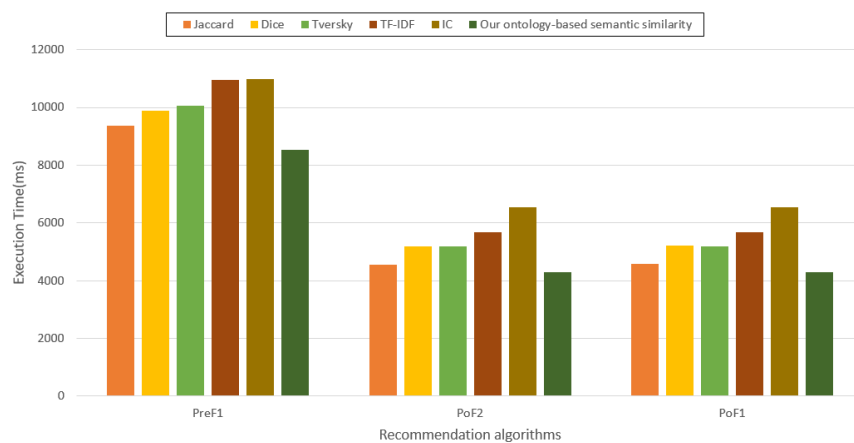


FIGURE 7.12: Execution time with different semantic similarities in a set of scientific collaborations.

In Figure 7.10, our ontology-based semantic similarity leads to the highest value of F1 when applied in PoF1 and PoF2. This indicates that our semantic similarity can enhance

F1 and thus generate more accurate context-aware collaborator recommendations. But it only achieve a medium value of F1 in PreF1. These varied values of F1 imply that the enhancing effect of our ontology-based semantic similarity depends on the recommendation algorithms, which is driven by the different phases of the semantic similarity in the algorithms (cf. Section 5.1).

Besides, when employing our ontology-based semantic similarity in the algorithms, their MAE results are always lower than those of Jaccard, Tversky, and IC. However, compared to Dice and TF-IDF, our semantic similarity produces slightly higher MAE results (except TF-IDF in PoF1). Considering that our semantic similarity brings significantly better results than Dice and TF-IDF in terms of F1, it is fair to conclude that our ontology-based semantic similarity can improve the accuracy of PoF1 and PoF2. When implemented in PreF1, our ontology-based semantic similarity has an average performance of accuracy, which is acceptable.

Lastly, our ontology-based semantic similarity evidently achieves the shortest execution time among all semantic similarities. This indicates that the time efficiencies of PreF1, PoF1, and PoF2 are augmented through the use of our semantic similarity.

7.3.6 Experiment with different collaborative filtering algorithms

In this experiment, another 2D recommendation technique Neural network-based Collaborative Filtering (NCF) (He et al., 2017) is employed in the context-aware recommendation algorithms: PreF1, PoF1, and PoF2 (see Algorithm 1). NCF can replace PMF in PreF1, PoF1, and PoF2 to generate 2D collaborator recommendations. Particularly, NCF also belongs to model-based CF approach and can be considered as an evolved version of MF with neural network architectures (He et al., 2017). In other words, NCF is a product of model-based CF and deep learning. Applying PMF and NCF in these algorithms gives us a chance to explore whether applying our ontology-based semantic similarity with different model-based CF techniques can influence the performances of these algorithms. Similar with other experiments, the accuracy and time efficiency are also evaluated through three metrics: F1, MAE and execution time. All the results are shown in Figures 7.13, 7.14, and 7.15.¹⁷

In summary, when either PMF or NCF is applied in the three algorithms (i.e. PreF1, PoF2 and PoF1), all values of F1 are higher than those of PMF and NCF themselves (represented as NCF/PMF in Figures 7.13, 7.14, and 7.15). This indicates that our ontology-based semantic similarity can increase F1 of PreF1, PoF2 and PoF1 whatever model-based CF technique is used. However, F1 values with NCF are much higher than those with PMF. This implies that the enhancement of F1 driven by our semantic similarity relates to the techniques used to generate 2D collaborator recommendations. Between NCF and PMF, our semantic similarity can attain greater F1 with NCF.

¹⁷In Figures 7.13, 7.14, and 7.15, PreF1 owns $z = 0$; PoF1 has $z = 1$; PoF2 refers to $z = 2$ (see Algorithm 1).

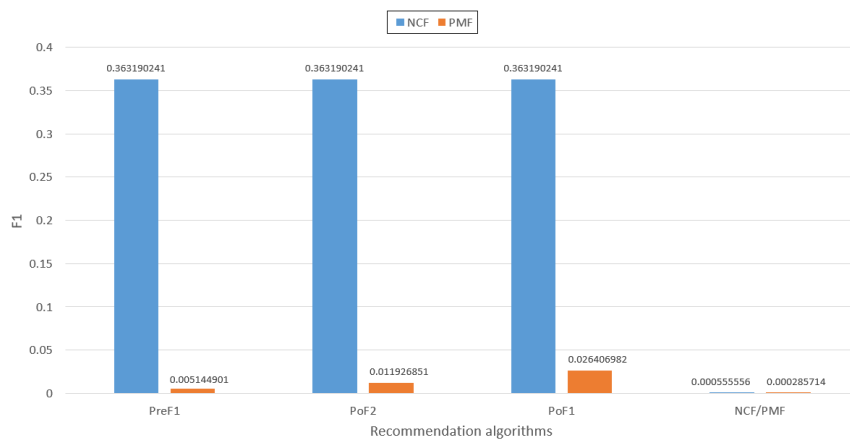


FIGURE 7.13: F1 with different 2D recommendation algorithms in a set of scientific collaborations.

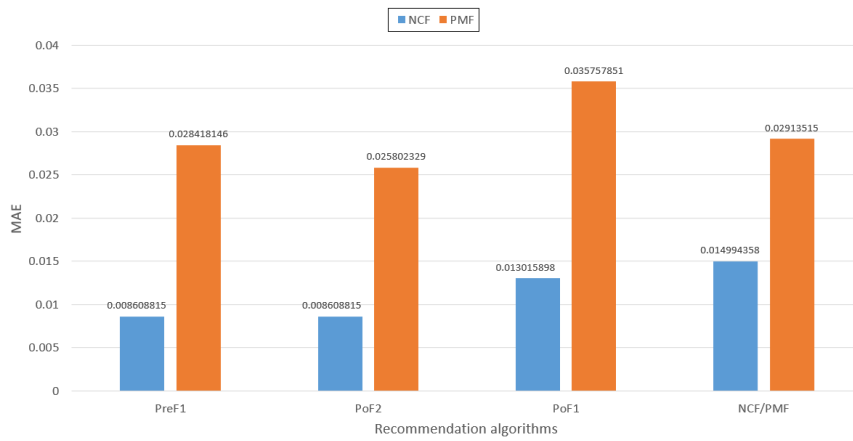


FIGURE 7.14: MAE with different 2D recommendation algorithms in a set of scientific collaborations.

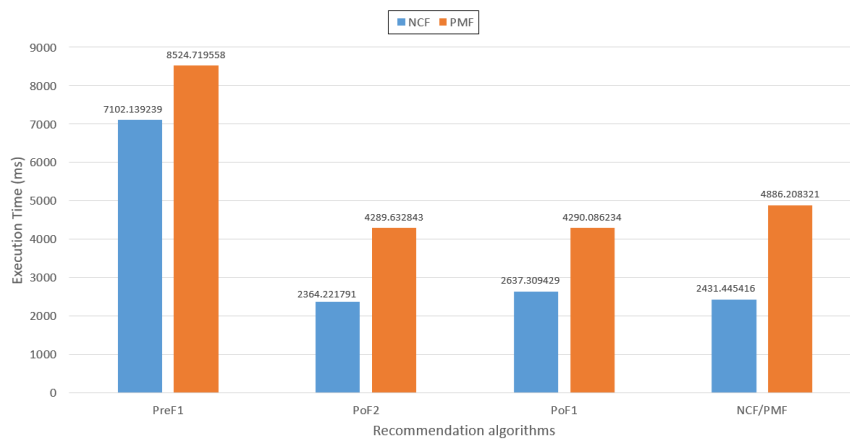


FIGURE 7.15: Execution time with different 2D recommendation algorithms in a set of scientific collaborations.

Meanwhile, compared with NCF/PMF, lower MAE are also reached when applying PMF and NCF in PreF1, PoF2 and PoF1 (except PMF in PoF1). This signifies that our ontology-based semantic similarity, while using either PMF or NCF in the algorithms, can reduce values of MAE. Similarly, the decreased MAE values with NCF are larger than those with PMF. This also means that the reduction of MAE is caused by our semantic similarity but linked to the techniques used to generate 2D collaborator recommendations. Both higher F1 and lower MAE suggest that our semantic similarity improves the accuracy of PreF1, PoF2, and PoF1. Only its enhancements are more evident with NCF than with PMF.

As for execution time, the results of PoF2, PoF1, and NCF/PMF are very near. On the contrary, the execution time of PreF1 is much longer than those of other algorithms, indicating a higher time complexity of PreF1.

7.4 Discussion

Utilizing a dataset of scientific collaborations, we conduct experiments on the performance of the context-aware collaborator recommendations algorithms: PreF1, PoF1, and PoF2¹⁸. Specifically, three metrics (i.e., F1, MAE, and execution time) are used to compare the performances for two terms: accuracy and time efficiency. We also apply two other algorithms in this dataset, namely CAMF (Baltrunas, Ludwig, and Ricci, 2011) and PMF (Mnih and Salakhutdinov, 2008). This allows us to compare the results of the algorithms following PreF and PoF methods with those of an algorithm of CM method and of a 2D recommendation algorithm (Adomavicius and Tuzhilin, 2011).¹⁹ Besides, PMF and NCF (He et al., 2017) are utilized to analyze the influences of applying our ontology-based semantic similarity with different model-based CF techniques to generate context-aware collaborator recommendations.

Based on the results of the experiments, the following can be concluded:

- PreF1 outperforms PMF and CAMF in F1 and MAE, but suffers in execution time due to its higher time complexity. In particular, the advantages of PreF1 in MAE can be influenced by different datasets.
- PoF1 beats all other algorithms in F1 and has an intermediate performance in execution time. But PoF1 performs worst in MAE due to the calculation of the adjusted ratings.
- The difference between PoF2, CAMF and PMF in execution time is minor. Also, PoF2 has better performance than CAMF and PMF in F1 and MAE.

¹⁸Here, PreF1, PoF1, and PoF2 have different values of z (see Algorithm 1). PreF1 owns $z = 0$; PoF1 has $z = 1$; PoF2 refers to $z = 2$.

¹⁹These algorithms are respectively known as PreF1 (based on PreF method), PoF1, PoF2 (based on PoF method), CAMF (based on CM method), and PMF (i.e. a 2D recommendation algorithm).

- The ontology-based semantic similarity (cf. Section 5.4) can shorten execution time of PreF1, PoF1, and PoF2. It also leads to higher accuracy in PoF1 and PoF2, but shows no improvement of accuracy in PreF1.
- Calculating our ontology-based semantic similarity in PreF1, PoF1, and PoF2 can produce context-aware collaborator recommendations with higher F1 and lower MAE, regardless of applying NCF or PMF. However, its positive effect on execution time isn't very obvious.

Besides, the ontology-based semantic similarity enables us to deal with the collaboration context in the context-aware collaborator recommendation generation processes. This type of context is never considered or discussed in previous studies on such recommendations. These previous studies usually focus on either user context (e.g., Liu, Xie, and Chen (2018)) or item context (e.g., Xu et al. (2010)). None of the previous studies takes into account users and items together. With the collaboration context, users and items are considered jointly in collaborations. This deepens the comprehension of both users and items in context-aware collaborator recommendations.

Also, this work has some limitations. The recommendations produced by PreF1 and PoF2 may have a serendipity problem (De Gemmis et al., 2015): the recommended collaborators are not 'surprising' to users. This leads to insufficient diversity of recommendations: it is often the same collaborators that are recommended to users. Based on PreF1 and PoF2 (see Algorithm 1), each collaborator recommended to the user must have involved in collaborations with high similarities. Such collaborators are obvious to facilitate users' collaborations, but not the good ones. The collaborators have not involved in the same collaborations with the user should also be included in the two algorithms, which may result in surprising and diverse recommendations.

7.5 Chapter summary

In this chapter, our work is mainly about the last question listed in Section 1.1.3: are our context-aware collaborator recommendations algorithms relevant in terms of accuracy and time efficiency?

Firstly, a dataset of scientific collaborations are applied to conduct experiments (see Table 7.5) on the performance of the context-aware collaborator recommendations algorithms: PreF1, PoF1, and PoF2 (cf. Section 5.5). Based on the results of these experiments, each algorithm has its advantages and disadvantages under different conditions. Generally, PreF1 has medium accuracy but suffers in time efficiency due to its higher time complexity. And PoF1 reaches better accuracy and time efficiency. PoF2 can lead us to the best accuracy and time efficiency. Therefore, PoF2 is our preferred choice when generating context-aware collaborator recommendations, except when K is very low. With small values of K , we choose

TABLE 7.5: Summary of experiments.

Experiments	Parameters	Better algorithms to choose
Experiment with different volume of training collaborations	Percentage of training collaborations	PoF2
Experiment with different number of generated recommendations	Value of K	PoF1 (if K is small) PoF2 (otherwise)
Experiment with different degree of closeness between collaborations	Set of scientific collaborations	PoF2
Experiment with different semantic similarities	Applied semantic similarity in algorithms	Our ontology-based semantic similarity in PreF1, PoF1, or PoF2
Experiment with different model-based CF techniques	Employed model-based CF technique in algorithms	PoF2

PoF1 because of its extremely high F1. Besides, compared to other semantic similarities, our ontology-based semantic similarity enhances both accuracy and time efficiency in these algorithms. Its ability to improve accuracy is unrelated to the applied model-based CF technique.

Part IV

Conclusion, Perspectives and Future Work

Chapter 8

Conclusion, perspectives and future

Work

This chapter is intended to conclude our work. It also presents some perspectives and possible future work of this thesis.

8.1 Conclusion

Collaborations, especially the successful ones, enable people to obtain more beneficial results compared to working individually. However, successful collaboration is often challenging, as it can be influenced by different factors (e.g., goal, resources) (Patel, Pettitt, and Wilson, 2012). Notably, collaborator is a significant impacting factor, which is essential in collaborations. Consequently, we focus on recommending appropriate collaborators to people, particularly when they are collaborating in web-based Collaborative Working Environments (CWEs). Integrating various collaborative tools (e.g. instant messaging and resource management tools), a web-based CWE allows people to overcome the limits of geographical distances between them during collaborations. However, a lot of information contained in web-based CWEs that can influence people's needs of collaborators is not considered while recommending collaborators, resulting irrelevant collaborator recommendations. Thus, to generate relevant collaborator recommendations in web-based CWEs, this thesis attempts to address four main questions, listed below.

- Q1. What is collaboration context and how to model it?
- Q2. How to process and apply the collaboration context in algorithms for generating context-aware collaborator recommendations?
- Q3. How to implement and integrate the collaboration context model into web-based CWEs?
- Q4. Are our context-aware collaborator recommendations algorithms relevant in terms of accuracy and time efficiency?

To solve Q1, we defined collaboration context and constructed a collaboration context ontology, MCC. MCC is developed upon an architecture of context models, which is formed by three interrelated concepts: contextual information, contextual factor, and contextual dimension. Particularly, MCC contains eight contextual dimensions (i.e., Goal, Collaborator, Activity, Resource, Time, Location, Relation, and Satisfaction), which can be used to describe a collaboration and its context. Each dimension includes several relevant contextual factors and their values (i.e., contextual information). These contextual dimensions, factors, and information permits us to clarify the characteristics of collaboration and its context in different phases.

In the collaboration context ontology MCC, any information is specified through semantic 3-uples $\langle \text{Subject, Predicate, Object} \rangle$. Particularly, a collaboration is represented by a collection of semantic 3-uples with a unique subject. With such expressions, we need to figure out how to handle and integrate semantic 3-uples into context-aware collaborator recommendation generation processes, leading us to the answer to Q2. Therefore, we developed an ontology-based semantic similarity, which can not only to compare the common and non-common objects in two collections of semantic 3-uples, but also to assess the amount of information provided by a specific predicate. This ontology-based semantic similarity enables us to develop three different context-aware collaborator algorithms (i.e., PreF1, PoF1, and PoF2), which are based on two existing methods of incorporating context into recommendation processes: contextual pre-filtering and post-filtering.

Next, to address the question Q3, a web-based CWE is considered as a collaborative System of Information Systems (SoIS). This permits us to consider an architecture of a collaborative SoIS (Saleh and Abel, 2016) from the viewpoint of web-based CWEs. Based on this architecture, the collaboration context ontology MCC can be implemented as an ontological knowledge base into web-based CWEs. It can process heterogeneous information from different collaborative tools and manage it within the context of users' collaborations. This then enables better information interoperability between heterogeneous tools and help users analyze their collaborations, thus improving information access and management during collaborations. Furthermore, a web-based CWE prototype, *MEMORAe* CWE, is built following the architecture. The main functionalities of this prototype were explained and tested by real users (i.e., students and teachers of a course at University of Technology of Compiegne).

Finally, we utilized a dataset of scientific collaborations for the last question Q4. This dataset well fits the needs of our experiments for several reasons:

- In this dataset, an academic article can be considered as a scientific collaboration. The detailed information of an article can be used to characterize the situation of a corresponding scientific collaboration over a given period of time, thus belonging to the context of this collaboration.

- This dataset is extracted from multiple academic bibliographies: DBLP¹, Microsoft Academic Graph (MAG)², and AMiner³. Each bibliography refers to a collaborative tools integrated in a web-based CWE. This corresponds to MCC that the collaboration context comes from different collaborative tools within a web-based CWE.
- From the aspect of volume, this dataset is able to measure the performance of the three context-aware collaborator recommendation algorithms (i.e., PreF1, PoF1, and PoF2), which employ our developed ontology-based semantic similarity in different phases to generate recommendations.
- In this dataset, we are able to apply other context-aware recommendation algorithms and 2D recommendation algorithms. This allows us to compare results of these algorithms with those of the context-aware collaborator recommendation algorithms. Specifically, we utilize F1, Mean Absolute Error (MAE), and execution time to analyze the performances of different algorithms from two aspects: accuracy and time efficiency.

Based on the results of our experiments, we obtained the advantages and disadvantages of each context-aware collaborator recommendation algorithm. Specifically, PreF1 is good at accuracy (i.e., F1 and MAE), but suffers in time efficiency (i.e., execution time) due to its higher time complexity; PoF1 is the best algorithm in F1 and has an intermediate performance in execution time, but performs worst in MAE due to the calculation of the adjusted ratings; PoF2 has excellent performances in both accuracy (i.e., F1 and MAE) and time efficiency (i.e., execution time). This indicates that PoF2 is our preferred choice to generate context-aware collaborator recommendations, except when the number of generated recommendations is very low. In such cases, we choose PoF1 because of its extremely high values of F1. Overall, all results signify that with our ontology-based semantic similarity, PreF1, PoF1, and PoF2 algorithms can enhance their performances, either in terms of accuracy or time efficiency, or both.

8.2 Perspectives and Future Work

Generating context-aware collaborator recommendations in web-based CWEs is a never-ending work. It can always be improved. In our opinions, there are several perspectives from which this work can continue. This section describes two different aspects of perspectives: short-term and long-term future work.

Particularly, short-term future work contains:

¹<https://dblp.org/>

²<https://academic.microsoft.com/home>

³<https://www.aminer.org/>

- In the carried out experiments, we changed many parameters to test and evaluate the performances of the context-aware collaborator recommendation algorithms: PreF1, PoF1, and PoF2. However, we only utilize two different 2D recommendation techniques (i.e., PMF and NCF) in these algorithms. It would be useful to apply other 2D recommendation techniques in these algorithms and compare with these results. This will help us investigate why applying our developed ontology-based semantic similarity with different 2D recommendation techniques in PreF1, PoF1, and PoF2 algorithms can influence their performances and how to improve the performances by employing other 2D recommendation techniques.
- Besides, the dataset we utilized is about scientific collaborations and extracted from multiple academic bibliographies. But it does not have any information related to the contextual dimension: Satisfaction. It would be interesting for us to apply a dataset with information related to all the contextual dimensions of collaboration. Such datasets should contain information on satisfactions, such as satisfaction with collaborations and with the generated recommendations. The results from these datasets might help us to determine whether the performances of the context-aware collaborator recommendation algorithms would be influenced by the completeness of the collaboration context in datasets.
- It would be worth building a context-aware collaborator recommender system in the prototype of web-based CWEs, *MEMORAe CWE*. If such a system could be developed, real users would be able to choose whether they want to receive context-aware collaborator recommendations and evaluate the performance of our algorithms. Meanwhile, this will allow us to obtain a dataset containing real users' information in a web-based CWE. With this dataset, we could carry out the experiments, which are same as those we performed in this thesis. This dataset could make the results of these experiments more complete and convincing.

Long-term future work includes:

- In the *MEMORAe* approach, Wang (2016) defined a competency module to measure people's competencies based on their activities. Such information can be used to describe the situation of a collaboration, thus belonging to the collaboration context. This implies that we can enrich the collaboration context ontology in this thesis by reusing the competency module (Wang, 2016), giving us more information of people's collaborations and their contexts.
- Based on the information containing in the collaboration context ontology, we can develop and realize other functionalities in web-based CWEs. For instance, after users complete or finish a collaboration in a web-based CWE, their profiles can be updated considering the information of this collaboration and its context.

- Further, the algorithm used to context-aware collaborator recommendations could be chosen according to people's requests. For example, based on the numbers of collaborator recommendations requested, we can select an algorithm from PoF1 and PoF2 to generate context-aware collaborator recommendations for people in web-based CWEs.
- Our ontology-based semantic similarity is used to compare two collaborations and their contexts. It also can serve to compare two other instances in the collaboration context ontology, such as resources. This implies that we might employ this ontology-based semantic similarity to generate other recommendations in web-based CWEs, such as context-aware resource recommendations. This will reveal the importance of our ontology-based semantic similarity in generating context-aware recommendations.
- In the collaboration context ontology, a collaboration goal consists of several concepts and their relationships (e.g., hierarchical relationships), which are shown respectively as nodes and links of a semantic map in the prototype *MEMORAe* CWE. These concepts and their relationships can be considered and/or reasoned to generate more specific collaborator recommendations. For instance, during collaborations, we can recommend collaborators to people for certain related concepts.

Publications

International Publications

- Li Siying, Abel Marie-Hélène, and Negre Elsa. "A collaboration context ontology to enhance human-related collaboration into Industry 4.0." In *Cognition, Technology & Work*, accepted.
- Li Siying, Abel Marie-Hélène, and Negre Elsa. "Ontology-based semantic similarity in generating context-aware collaborator recommendations." In 2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD), regular paper, accepted.
- Li Siying, Abel Marie-Hélène, and Negre Elsa. "A Collaborative Working Environment as an ontology-based collaborative System of Information Systems." In 2020 IEEE International Conference on Systems, Man and Cybernetics (SMC), regular paper, pp.800-805. Toronto, Canada, Oct 2020.
- Li Siying, Abel Marie-Hélène, and Negre Elsa. "Managing and recommending resources in web-based collaborative working environments." In 2020 IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), short paper, pp.287-290. Bayonne, France, Sep 2020.
- Li Siying, Abel Marie-Hélène, and Negre Elsa. "Towards a collaboration context ontology." In 2019 IEEE 23rd International Conference on Computer Supported Cooperative Work in Design (CSCWD), regular paper, pp.93-98. Porto, Portugal, May 2019.
- Li Siying, Abel Marie-Hélène, and Negre Elsa. "Using User Contextual Profile for Recommendation in Collaborations." In Rii Forum 2019 Proceedings, regular paper, pp.93-98. Rome, Italy, April 2019.
- Li Siying, Abel Marie-Hélène, and Negre Elsa. "Contact and Collaboration Context Model." In 2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI), regular paper, pp.1-6. Palermo, Italy, Sep 2018.

National Publications

- Li Siying, Abel Marie-Hélène, and Negre Elsa. "MEMORAe-CWE : un système collaboratif de systèmes d'information à base d'ontologies" In 31ème Journées Francophones d'Ingénierie des Connaissances, IC 2020, regular paper, pp.56-71. Angers, France, Jun 2020.
- Li Siying, Abel Marie-Hélène, and Negre Elsa. "Modèle de contexte de collaboration: pour qui, pourquoi, comment?." In 29es Journées Francophones d'Ingénierie des Connaissances, IC 2018, regular paper, pp.229-243. Nancy, France, Jul 2018.

Appendix A

The detailed gradient descent algorithm

The gradients of $\mathcal{L}(U, V)$ (see Equation 5.4) are (Mnih and Salakhutdinov, 2008; Lemaréchal, 2012):

$$\begin{cases} \frac{\partial \mathcal{L}(U, V)}{\partial U_i} = -I_{ij}(R_{ij} - U_i^T V_j) V_j + \lambda_U U_i \\ \frac{\partial \mathcal{L}(U, V)}{\partial V_j} = -I_{ij}(R_{ij} - U_i^T V_j) U_i + \lambda_V V_j \end{cases} \quad (\text{A.1})$$

PMF gradient descent algorithm requires input data: d (the number of latent factors), λ_U, λ_V (regularization parameters defined in Equation 5.4), R (rating matrix of $m \times n$), η (the learning rate in gradient descent algorithm), and $Step$ (predefined maximum iteration steps).

Initially, two matrices U of $m \times d$ and V of $n \times d$ are given. Then a loop starts to find a local minimum $\mathcal{L}(U, V)$ based on Equation 5.4. Each step applies the gradients in Equation A.1 to update U_i in U and V_j in V (Equation A.2)¹.

$$\begin{cases} U_i^{(t+1)} = U_i^{(t)} - \eta * \frac{\partial \mathcal{L}(U, V)}{\partial U_i} \\ V_j^{(t+1)} = V_j^{(t)} - \eta * \frac{\partial \mathcal{L}(U, V)}{\partial V_j} \end{cases} \quad (\text{A.2})$$

Using these gradients, the algorithm iteratively updates the estimations of U and V until convergence or the maximum iteration steps. Finally, the outputs include updated U and V , and the minimum $\mathcal{L}(U, V)$. The pseudo codes of PMF gradient descent algorithm is shown in Algorithm 4.

¹<https://towardsdatascience.com/probabilistic-matrix-factorization-b7852244a321>

Algorithm 4: Algorithm of PMF

Input: The number of latent factor d , the learning rate η , regularization parameters λ_U, λ_V , the max iteration $Step$, and the rating matrix R

Initialization: Initialize a random matrix for user matrix U and item matrix V

```

1 for  $t := 1, 2, \dots, Step$  do
2    $\mathcal{L}(U, V) := 0$ ;
3   for  $(i, j)$  in  $R$  do
4      $r := R_{ij}$ ;
5     make prediction  $pr := U_i^T * V_j$ ;
6     error  $e := r - pr$ ;
7      $U_i := U_i + \eta * (e * V_j - \lambda_V * U_i)$ ;
8      $V_j := V_j + \eta * (e * U_i - \lambda_V * V_j)$ ;
9      $\mathcal{L}(U, V) :=$ 
        $\mathcal{L}(U, V) + 0.5 * (e^2 + \lambda_U * \|U_i\|_{Fro}^2 + \lambda_V * \|V_j\|_{Fro}^2)$ 

```

Output: The minimum sum-of-squared-errors objective function $\mathcal{L}(U, V)$, user matrix U , and item matrix V .

Appendix B

The interfaces for accessing and manipulating instances

TABLE B.1: Interface for the instances.

		URL http://memorae.hds.utc.fr/api/abox /<Instance_Type>/<Instance_ID>
HTTP Operation	GET	Retrieve the representation of an instance
	PUT	Modify the representation of an instance (if Instance_ID exists) Create the representation of an instance (if Instance_ID does not exist)
	DELETE	Delete the representation of an instance

TABLE B.2: Interface for the lists of instances.

		URL http://memorae.hds.utc.fr/api/abox/<Instance_Type>
HTTP Operation	GET	Retrieve a list of instances
	POST	Create a new representation of an instance in a list
	DELETE	Delete a list of instances

Appendix C

The survey of MEMORAe CWE

MEMORAe CWE Survey

A short survey aims at understanding students' satisfactions and suggestions about the web-based collaborative working environment, MEMORAe CWE.

**Required*

1. What is your age? *

2. Your gender *

Mark only one oval.

- Male
 Female
 Prefer not to say
 Other: _____

3. What TD group are you from? *

Mark only one oval.

- TD 1
 TD 2
 TD 3
 TD 5
 TD 7
 Other: _____

4. MEMORAe CWE can be used to help you understand the notions in your courses. *

Mark only one oval.

0	1	2	3	4	5	6	7	8	9	10	
Not agree at all	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Totally agree

5. MEMORAe CWE can help you revise your courses. *

Mark only one oval.

0 1 2 3 4 5 6 7 8 9 10

Not agree at all Totally agree

6. MEMORAe CWE can organize resources of your courses. *

Mark only one oval.

0 1 2 3 4 5 6 7 8 9 10

Not agree at all Totally agree

7. MEMORAe CWE can allow you to work together with others (students and professors). *

Mark only one oval.

0 1 2 3 4 5 6 7 8 9 10

Not agree at all Totally agree

8. MEMORAe CWE is useful for you courses during this semester. *

Mark only one oval.

0 1 2 3 4 5 6 7 8 9 10

Not agree at all Totally agree

MEMORAe CWE's functionalities

9. How often do you use MEMORAe CWE in your courses? *

Mark only one oval.

- Several times per week
- Several times per month
- Several times every 2-3 months
- Several times per semester
- Never
- Other: _____

10. What type of sharing spaces have you used? (multiple choice) *

Tick all that apply.

- Sharing space of NF01 group
- Sharing space of TD group
- Sharing space of private group
- Sharing space of project(small working) group

11. What activities have you done in these sharing spaces? (multiple choice) *

Tick all that apply.

- Uploading/Adding resources
 Consulting resources
 Sharing resources
 Voting resources
 Commenting resources
 Annotating resources

Other: _____

12. Have you added or shared same resources in different sharing spaces? *

Mark only one oval.

- Yes
 No

13. Can consulting activity traces helps you to identify people related to a subject? (Here, activities includes uploading/adding, consulting, sharing, voting, commenting and annotating resources.) *

Mark only one oval.

- Yes
 Maybe
 No

14. If MEMORAe CWE can suggest people related to a subject based on their activity traces, are you interested in this functionality? *

Mark only one oval.

0	1	2	3	4	5	6	7	8	9	10	
Not interested at all	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Very interested

15. Have you used integrated collaborative tools in MEMORAe CWE? (multiple choice) *

Tick all that apply.

- Google Search
 Web Link
 Youtube
 File
 Google Drive

16. Would you like have access to other collaborative tools? What are they?

17. Have you used other similar collaborative platforms like MEMORAe CWE? *

Mark only one oval.

- Yes
 No
 Maybe

18. What are these collaborative platforms?

19. As a learning support, how would you rate for MEMORAe CWE? *

Mark only one oval.

	0	1	2	3	4	5	6	7	8	9	10	
Not useful at all	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Very useful

20. As a digital collaborative support, how would you rate for MEMORAe CWE? *

Mark only one oval.

	0	1	2	3	4	5	6	7	8	9	10	
Not useful at all	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Very useful

21. Do you have any suggestions and comments of MEMORAe CWE?

Bibliography

- Adomavicius, Gediminas and Dietmar Jannach (2014). “Preface to the special issue on context-aware recommender systems”. In: *User Modeling and User-Adapted Interaction* 24.1-2, pp. 1–5.
- Adomavicius, Gediminas and Alexander Tuzhilin (2005). “Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions”. In: *IEEE Transactions on Knowledge & Data Engineering* 6, pp. 734–749.
- (2011). “Context-aware recommender systems”. In: *Recommender systems handbook*. Springer, pp. 217–253.
- Adomavicius, Gediminas et al. (2005). “Incorporating contextual information in recommender systems using a multidimensional approach”. In: *ACM Transactions on Information Systems (TOIS)* 23.1, pp. 103–145.
- Afsarmanesh, Hamideh et al. (2009). “Modeling and management of information supporting functional dimension of collaborative networks”. In: *Transactions on Large-Scale Data and Knowledge-Centered Systems I*. Springer, pp. 1–37.
- Ameur, Mohamed Ali Ben et al. (2017). “Recommendation of Pedagogical Resources within a Learning Ecosystem”. In: *The 9th International Conference on Management of Digital EcoSystems*.
- Anand, Sarabjot Singh and Bamshad Mobasher (2007). “Introduction to intelligent techniques for web personalization”. In: *ACM Transactions on Internet Technology (TOIT)* 7.4, p. 18.
- Assaad, Mohamad Ali, Reine Talj, and Ali Charara (July 2016). “A view on Systems of Systems (SoS)”. In: *20th World Congress of the International Federation of Automatic Control (IFAC WC 2017) - special session*. Toulouse, France. URL: <https://hal.archives-ouvertes.fr/hal-01741416>.
- Atrash, Ala (2015). “Modeling a system of expertise capitalization to support organizational learning within small and medium-sized enterprises”. PhD thesis. Université de Technologie de Compiègne.
- Atrash, Ala, Marie-Hélène Abel, and Claude Moulin (2014). “Supporting organizational learning with collaborative annotation”. In: *International Conference on Knowledge Management and Information Sharing*, pp. 237–244.
- Azarnoush, Hamed et al. (2006). “Towards optimization of a real-world robotic-sensor system of systems”. In: *2006 World Automation Congress*. IEEE, pp. 1–8.

- Bafoutsou, Georgia and Gregoris Mentzas (2002). "Review and functional classification of collaborative systems". In: *International journal of information management* 22.4, pp. 281–305.
- Balabanović, Marko and Yoav Shoham (1997). "Fab: content-based, collaborative recommendation". In: *Communications of the ACM* 40.3, pp. 66–72.
- Baltrunas, Linas, Bernd Ludwig, and Francesco Ricci (2011). "Matrix factorization techniques for context aware recommendation". In: *Proceedings of the fifth ACM conference on Recommender systems*. ACM, pp. 301–304.
- Baltrunas, Linas et al. (2012). "Context relevance assessment and exploitation in mobile recommender systems". In: *Personal and Ubiquitous Computing* 16.5, pp. 507–526.
- Batet, Montserrat, David Sánchez, and Aida Valls (2011). "An ontology-based measure to compute semantic similarity in biomedicine". In: *Journal of biomedical informatics* 44.1, pp. 118–125.
- Bazire, Mary and Patrick Brézillon (2005). "Understanding context before using it". In: *International and Interdisciplinary Conference on Modeling and Using Context*. Springer, pp. 29–40.
- Bentley, Richard et al. (1997). "Basic support for cooperative work on the World Wide Web". In: *International journal of human-computer studies* 46.6, pp. 827–846.
- Bettini, Claudio et al. (2010). "A survey of context modelling and reasoning techniques". In: *Pervasive and Mobile Computing* 6.2, pp. 161–180.
- Billsus, Daniel, Michael J Pazzani, et al. (1998). "Learning Collaborative Information Filters." In: *Icml*. Vol. 98, pp. 46–54.
- Blei, David M, Andrew Y Ng, and Michael I Jordan (2003). "Latent dirichlet allocation". In: *Journal of machine Learning research* 3.Jan, pp. 993–1022.
- Bowen, Ryan M and Ferat Sahin (2010). "A net-centric xml based system of systems architecture for human tracking". In: *2010 5th International Conference on System of Systems Engineering*. IEEE, pp. 1–6.
- Breese, John S, David Heckerman, and Carl Kadie (1998). "Empirical analysis of predictive algorithms for collaborative filtering". In: *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., pp. 43–52.
- Bridge, Derek et al. (2005). "Case-based recommender systems". In: *The Knowledge Engineering Review* 20.3, pp. 315–320.
- Briggs, Robert O et al. (2009). "A seven-layer model of collaboration: Separation of concerns for designers of collaboration systems". In: *ICIS 2009 Proceedings*, p. 26.
- Brown, Peter J, John D Bovey, and Xian Chen (1997). "Context-aware applications: from the laboratory to the marketplace". In: *IEEE personal communications* 4.5, pp. 58–64.
- Bruneel, Johan, Pablo d'Este, and Ammon Salter (2010). "Investigating the factors that diminish the barriers to university–industry collaboration". In: *Research policy* 39.7, pp. 858–868.
- Burke, Robin (2000). "Knowledge-based recommender systems". In: *Encyclopedia of library and information systems* 69.Supplement 32, pp. 175–186.

- Burke, Robin (2002). “Hybrid recommender systems: Survey and experiments”. In: *User modeling and user-adapted interaction* 12.4, pp. 331–370.
- Camarinha-Matos, LM and H Afsarmanesh (2006). “Collaborative networks: Value creation in a knowledge society”. In: *Proceedings of PROLAMAT 2006 IFIP international conference on knowledge enterprise: Intelligent Strategies in Product Design, Manufacturing, and Management*. Springer, pp. 26–40.
- Camarinha-Matos, Luis M and Hamideh Afsarmanesh (2005). “Collaborative networks: a new scientific discipline”. In: *Journal of intelligent manufacturing* 16.4-5, pp. 439–452.
- (2008). *Collaborative networks: Reference modeling*. Springer Science & Business Media.
- Cantador, Iván, Alejandro Bellogín, and Pablo Castells (2008). “A multilayer ontology-based hybrid recommendation model”. In: *Ai Communications* 21.2-3, pp. 203–210.
- Carrer-Neto, Walter et al. (2012). “Social knowledge-based recommender system. Application to the movies domain”. In: *Expert Systems with applications* 39.12, pp. 10990–11000.
- Carreras, MA Martínez and AF Gómez Skarmeta (2006). “Towards interoperability in collaborative environments”. In: *2006 International Conference on Collaborative Computing: Networking, Applications and Worksharing*. IEEE, pp. 1–5.
- Carstensen, Peter H and Kjeld Schmidt (1999). “Computer supported cooperative work: New challenges to systems design”. In: *In K. Itoh (Ed.), Handbook of Human Factors*. Cite-seer.
- Chen, Guanling and David Kotz (2000). “A survey of context-aware mobile computing research”. In: *Dartmouth Computer Science Technical Report TR2000-381*.
- Chen, Rung-Ching et al. (2012). “A recommendation system based on domain ontology and SWRL for anti-diabetic drugs selection”. In: *Expert Systems with Applications* 39.4, pp. 3995–4006.
- Cimiano, Philipp (Jan. 2006). *Ontology Learning and Population from Text — Algorithms, Evaluation and Applications*. DOI: [10.1007/978-0-387-39252-3](https://doi.org/10.1007/978-0-387-39252-3).
- Dahmann, Judith S and Kristen J Baldwin (2008). “Understanding the current state of US defense systems of systems and the implications for systems engineering”. In: *2008 2nd Annual IEEE Systems Conference*. IEEE, pp. 1–7.
- Das, Abhinandan S et al. (2007). “Google news personalization: scalable online collaborative filtering”. In: *Proceedings of the 16th international conference on World Wide Web*. ACM, pp. 271–280.
- De Gemmis, Marco et al. (2015). “An investigation on the serendipity problem in recommender systems”. In: *Information Processing & Management* 51.5, pp. 695–717.
- Deparis, Etienne (2013). “Création de nouvelles connaissances décisionnelles pour une organisation via ses ressources sociales et documentaires”. PhD thesis. Université de Technologie de Compiègne.

- Desrosiers, Christian and George Karypis (2011). “A comprehensive survey of neighborhood-based recommendation methods”. In: *Recommender systems handbook*. Springer, pp. 107–144.
- Dey, Anind K (2001). “Understanding and using context”. In: *Personal and ubiquitous computing 5.1*, pp. 4–7.
- Dey, Anind K, Gregory D Abowd, and Daniel Salber (2001). “A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications”. In: *Human-Computer Interaction 16.2-4*, pp. 97–166.
- Dice, Lee R (1945). “Measures of the amount of ecologic association between species”. In: *Ecology 26.3*, pp. 297–302.
- DiMario, Michael J, John T Boardman, and Brian J Sauser (2009). “System of systems collaborative formation”. In: *IEEE Systems Journal 3.3*, pp. 360–368.
- DoD, ATL (2008). “System of Systems Systems Engineering Guide: Considerations for Systems Engineering In a System of Systems Environment”. In: *Deputy Undersecretary of Defense (Acquisition and Technology), Office of the Undersecretary of Defense (Acquisition, Technology, and Logistics)*.
- Doherty, Meghan et al. (2015). “Factors of Successful Collaboration: Oregon’s Watershed Councils as Collaborative Systems”. In:
- Dou, Dejing, Hao Wang, and Haishan Liu (2015). “Semantic data mining: A survey of ontology-based approaches”. In: *Proceedings of the 2015 IEEE 9th international conference on semantic computing (IEEE ICSC 2015)*. IEEE, pp. 244–251.
- Du, Yu et al. (July 2019). “Apports des ontologies aux systèmes de recommandation : état de l’art et perspectives”. In: *30es Journées Francophones d’Ingénierie des Connaissances, IC 2019*. Toulouse, France, pp. 64–77.
- Felfernig, Alexander and Robin Burke (2008). “Constraint-based recommender systems: technologies and research issues”. In: *Proceedings of the 10th international conference on Electronic commerce*. ACM, p. 3.
- Felfernig, Alexander et al. (2006). “An integrated environment for the development of knowledge-based recommender applications”. In: *International Journal of Electronic Commerce 11.2*, pp. 11–34.
- (2011). “Developing constraint-based recommenders”. In: *Recommender systems handbook*. Springer, pp. 187–215.
- Feng, Ling, Peter MG Apers, and Willem Jonker (2004). “Towards context-aware data management for ambient intelligence”. In: *International conference on database and expert systems applications*. Springer, pp. 422–431.
- Ferdousi, Zahra Vahidi, Elsa Negre, and Dario Colazzo (May 2017). “Context Factors in Context-Aware Recommender System”. In: *AISR 2017 Atelier interdisciplinaire sur les systèmes de recommandation*. Paris, France. URL: <https://hal.archives-ouvertes.fr/hal-01643553>.

- Franklin, David and Joshua Flaschbart (1998). "All gadget and no representation makes jack a dull environment". In: *Proceedings of the AAAI 1998 Spring Symposium on Intelligent Environments*, pp. 155–160.
- Gorod, Alex et al. (2007). "System of systems management: A network management approach". In: *2007 IEEE International Conference on System of Systems Engineering*. IEEE, pp. 1–5.
- Goutte, Cyril and Eric Gaussier (2005). "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation". In: *European conference on information retrieval*. Springer, pp. 345–359.
- Gruber, Thomas R (1993). "A translation approach to portable ontology specifications". In: *Knowledge acquisition 5.2*, pp. 199–220.
- Hara, Noriko et al. (2003). "An emerging view of scientific collaboration: Scientists' perspectives on collaboration and factors that impact collaboration". In: *Journal of the American Society for Information science and Technology* 54.10, pp. 952–965.
- He, Xiangnan et al. (2017). "Neural collaborative filtering". In: *Proceedings of the 26th international conference on world wide web*, pp. 173–182.
- Held, Albert, Sven Buchholz, and Alexander Schill (2002). "Modeling of context information for pervasive computing applications". In: *Proceedings of SCI*, pp. 167–180.
- Henricksen, Karen, Jadwiga Indulska, and Andry Rakotonirainy (2002). "Modeling context information in pervasive computing systems". In: *International Conference on Pervasive Computing*. Springer, pp. 167–180.
- Hofte, Gerard Henri ter (1998). "Working apart together: Foundations for component groupware". In: pp. 34–52.
- Hong, Jong-yi, Eui-ho Suh, and Sung-Jin Kim (2009). "Context-aware systems: A literature review and classification". In: *Expert Systems with applications* 36.4, pp. 8509–8522.
- Jaccard, Paul (1901). "Étude comparative de la distribution florale dans une portion des Alpes et des Jura". In: *Bull Soc Vaudoise Sci Nat* 37, pp. 547–579.
- Jamshidi, Mo (2008). *Systems of systems engineering: principles and applications*. CRC press.
- Jannach, Dietmar et al. (2010). "Hybrid recommendation approaches". In: *Recommender Systems: An Introduction*. Cambridge University Press, 124–142. DOI: [10 . 1017 / CBO9780511763113.007](https://doi.org/10.1017/CBO9780511763113.007).
- Jin, Rong, Joyce Y Chai, and Luo Si (2004). "An automatic weighting scheme for collaborative filtering". In: *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 337–344.
- Kan, HY, Vincent G Duffy, and Chuan-Jun Su (2001). "An Internet virtual reality collaborative environment for effective product design". In: *Computers in Industry* 45.2, pp. 197–213.
- Karatzoglou, Alexandros et al. (2010). "Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering". In: *Proceedings of the fourth ACM conference on Recommender systems*. ACM, pp. 79–86.

- Karcanias, Nicos and Ali G Hessami (2011). "System of systems and emergence part 1: Principles and framework". In: *2011 Fourth International Conference on Emerging Trends in Engineering & Technology*. IEEE, pp. 27–32.
- Kofod-Petersen, Anders and Jörg Cassens (2005). "Using activity theory to model context awareness". In: *International Workshop on Modeling and Retrieval of Context*. Springer, pp. 1–17.
- Koren, Yehuda and Robert Bell (2015). "Advances in collaborative filtering". In: *Recommender systems handbook*. Springer, pp. 77–118.
- Koren, Yehuda, Robert Bell, and Chris Volinsky (2009). "Matrix factorization techniques for recommender systems". In: *Computer* 8, pp. 30–37.
- Kotlarsky, Julia and Ilan Oshri (2005). "Social ties, knowledge sharing and successful collaboration in globally distributed system development projects". In: *European Journal of Information Systems* 14.1, pp. 37–48.
- Laso Ballesteros, Isidro and Wolfgang Prinz (2008). *New Collaborative Working Environments 2020*. Report on industry-led FP7 consultations and 3rd Report of the Experts Group on Collaboration@Work.
- Lemaréchal, Claude (2012). "Cauchy and the gradient method". In: *Doc Math Extra* 251.254, p. 10.
- Li, Qiang, Marie-Hélène Abel, and Jean-Paul Barthès (2012). "Sharing working experience: Using a model of Collaborative Traces". In: *Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. IEEE, pp. 221–227.
- Lieberman, Henry and Ted Selker (2000). "Out of context: Computer systems that adapt to, and learn from, context". In: *IBM systems journal* 39.3.4, pp. 617–632.
- Liu, Zheng, Xing Xie, and Lei Chen (2018). "Context-aware academic collaborator recommendation". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, pp. 1870–1879.
- Lops, Pasquale, Marco De Gemmis, and Giovanni Semeraro (2011). "Content-based recommender systems: State of the art and trends". In: *Recommender systems handbook*. Springer, pp. 73–105.
- Lorenzi, Fabiana and Francesco Ricci (2003). "Case-based recommender systems: A unifying view". In: *IJCAI Workshop on Intelligent Techniques for Web Personalization*. Springer, pp. 89–113.
- Lucchi, Roberto, Michel Millot, and Christian Elfers (2008). "Resource oriented architecture and REST". In: *Assessment of impact and advantages on INSPIRE, Ispra: European Communities*.
- Ma, Hao et al. (2008). "Sorec: social recommendation using probabilistic matrix factorization". In: *Proceedings of the 17th ACM conference on Information and knowledge management*. ACM, pp. 931–940.

- Maier, Mark W (1998). “Architecting principles for systems-of-systems”. In: *Systems Engineering: The Journal of the International Council on Systems Engineering* 1.4, pp. 267–284.
- Man, Tong et al. (2015). “Context-adaptive matrix factorization for multi-context recommendation”. In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, pp. 901–910.
- Martínez-Carreras, M Antonia et al. (2007). “Designing a generic collaborative working environment”. In: *IEEE International Conference on Web Services (ICWS 2007)*. IEEE, pp. 1080–1087.
- Mattessich, Paul W and Barbara R Monsey (1992). *Collaboration: what makes it work. A review of research literature on factors influencing successful collaboration*. ERIC.
- McCarthy, John (1993). “Notes on formalizing context”. In:
- Melville, Prem, Raymond J Mooney, and Ramadass Nagarajan (2002). “Content-boosted collaborative filtering for improved recommendations”. In: *Aaai/iaai* 23, pp. 187–192.
- Middleton, Stuart E, Nigel R Shadbolt, and David C De Roure (2004). “Ontological user profiling in recommender systems”. In: *ACM Transactions on Information Systems (TOIS)* 22.1, pp. 54–88.
- Miller, Bradley N et al. (2004). “Movielens unplugged: Experiences with a recommender system on four mobile devices”. In: *People and Computers XVII—Designing for Society*. Springer, pp. 263–279.
- Mnih, Andriy and Ruslan R Salakhutdinov (2008). “Probabilistic matrix factorization”. In: *Advances in neural information processing systems*, pp. 1257–1264.
- Mohsin, Ahmad et al. (2019). “A Taxonomy of Modeling Approaches for Systems-of-Systems Dynamic Architectures: Overview and Prospects”. In: *arXiv preprint arXiv:1902.09090*.
- Munir, Kamran and M Sheraz Anjum (2018). “The use of ontologies for effective knowledge modelling and information retrieval”. In: *Applied Computing and Informatics* 14.2, pp. 116–126.
- Neale, Dennis C, John M Carroll, and Mary Beth Rosson (2004). “Evaluating computer-supported cooperative work: models and frameworks”. In: *Proceedings of the 2004 ACM conference on Computer supported cooperative work*. ACM, pp. 112–121.
- Negre, Elsa (2015). *Information and recommender systems*. John Wiley & Sons.
- (2017). “Prise en compte du contexte dans les systèmes de recommandations de requetes OLAP”. In: *EDA 2017: BI & Big Data*.
- (2018). “Vers une typologie de contexte pour les systèmes de recommandation”. In:
- Neo, Mai (2003). “Developing a collaborative learning environment using a web-based design”. In: *Journal of computer assisted learning* 19.4, pp. 462–473.
- Neto, Valdemar Vicente Graciano, Renata Araujo, and Rodrigo Pereira dos Santos (2017). “New challenges in the social web: Towards systems-of-information systems ecosystems”. In: *Anais do VIII Workshop sobre Aspectos da Interação Humano-Computador para a Web Social*. SBC, pp. 1–12.

- Ning, Xia and George Karypis (2011). “Slim: Sparse linear methods for top-n recommender systems”. In: *2011 IEEE 11th International Conference on Data Mining*. IEEE, pp. 497–506.
- Nunes, Ingrid and Dietmar Jannach (2017). “A systematic review and taxonomy of explanations in decision support and recommender systems”. In: *User Modeling and User-Adapted Interaction* 27.3-5, pp. 393–444.
- Oliveira, Isolina, Luis Tinoca, and Alda Pereira (2011). “Online group work patterns: How to promote a successful collaboration”. In: *Computers & Education* 57.1, pp. 1348–1357.
- Oord, Aaron Van den, Sander Dieleman, and Benjamin Schrauwen (2013). “Deep content-based music recommendation”. In: *Advances in neural information processing systems*, pp. 2643–2651.
- Ovaska, Kristian, Marko Laakso, and Sampsa Hautaniemi (2008). “Fast Gene Ontology based clustering for microarray experiments”. In: *BioData mining* 1.1, p. 11.
- Palmisano, Cosimo, Alexander Tuzhilin, and Michele Gorgoglione (2008). “Using context to improve predictive modeling of customers in personalization applications”. In: *IEEE transactions on knowledge and data engineering* 20.11, pp. 1535–1549.
- Panniello, Umberto and Michele Gorgoglione (2011). “Context-Aware Recommender Systems: A Comparison Of Three Approaches.” In: *DART@ AI* IA*.
- (2012). “Incorporating context into recommender systems: an empirical comparison of context-based approaches”. In: *Electronic Commerce Research* 12.1, pp. 1–30.
- Pascoe, Jason (1998). “Adding generic contextual capabilities to wearable computers”. In: *2nd international symposium on wearable computers*. Ieee Computer Soc, pp. 92–99.
- Patel, Harshada, Michael Pettitt, and John R Wilson (2012). “Factors of collaborative working: A framework for a collaboration model”. In: *Applied ergonomics* 43.1, pp. 1–26.
- Peckham, Joan and Fred Maryanski (1988). “Semantic data models”. In: *ACM Computing Surveys (CSUR)* 20.3, pp. 153–189.
- Petrakis, Euripides GM et al. (2006). “X-similarity: Computing semantic similarity between concepts from different ontologies.” In: *Journal of Digital Information Management* 4.4.
- Prinz, Wolfgang et al. (2006). “ECOSPACE—towards an integrated collaboration space for eProfessionals”. In: *2006 International Conference on Collaborative Computing: Networking, Applications and Worksharing*. IEEE, pp. 1–7.
- Rada, Roy et al. (1989). “Development and application of a metric on semantic nets”. In: *IEEE transactions on systems, man, and cybernetics* 19.1, pp. 17–30.
- Rajaraman, Anand and Jeffrey David Ullman (2011). *Mining of massive datasets*. Cambridge University Press.
- Raza, Shaina and Chen Ding (2019). “Progress in context-aware recommender systems—An overview”. In: *Computer Science Review* 31, pp. 84–97.
- Ricci, Francesco, Lior Rokach, and Bracha Shapira (2011). “Introduction to recommender systems handbook”. In: *Recommender systems handbook*. Springer, pp. 1–35.
- (2015). “Recommender systems: introduction and challenges”. In: *Recommender systems handbook*. Springer, pp. 1–34.

- Ristoski, Petar and Heiko Paulheim (2016). "Semantic Web in data mining and knowledge discovery: A comprehensive survey". In: *Journal of Web Semantics* 36, pp. 1–22.
- Rodden, Tom et al. (1998). "Exploiting context in HCI design for mobile systems". In: *Workshop on human computer interaction with mobile devices*. Citeseer, pp. 21–22.
- Roussey, Catherine et al. (2011). "An introduction to ontologies and ontology engineering". In: *Ontologies in Urban development projects*. Springer, pp. 9–38.
- Ryan, Nick S, Jason Pascoe, and David R Morse (1998). "Enhanced reality fieldwork: the context-aware archaeological assistant". In: *Computer applications in archaeology*. Tempus Reparatum.
- Salakhutdinov, Ruslan and Andriy Mnih (2008). "Bayesian probabilistic matrix factorization using Markov chain Monte Carlo". In: *Proceedings of the 25th international conference on Machine learning*. ACM, pp. 880–887.
- Saleh, Majd (2018). "Digital ecosystem towards a System of Information Systems". PhD thesis. Université de Technologie de Compiègne.
- Saleh, Majd and Marie-Hélène Abel (2016). "Moving from digital ecosystem to system of information systems". In: *2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. IEEE, pp. 91–96.
- San Martín-Rodríguez, Leticia et al. (2005). "The determinants of successful collaboration: a review of theoretical and empirical studies". In: *Journal of interprofessional care* 19.sup1, pp. 132–147.
- Sánchez, David et al. (2012). "Ontology-based semantic similarity: A new feature-based approach". In: *Expert systems with applications* 39.9, pp. 7718–7728.
- Sarwar, Badrul et al. (2001). "Item-based collaborative filtering recommendation algorithms". In: *Proceedings of the 10th international conference on World Wide Web*, pp. 285–295.
- Schilit, Bill N, Norman Adams, Roy Want, et al. (1994). *Context-aware computing applications*. Xerox Corporation, Palo Alto Research Center.
- Schmidt, Albrecht, Michael Beigl, and Hans-W Gellersen (1999). "There is more to context than location". In: *Computers & Graphics* 23.6, pp. 893–901.
- Shan, Hanhuai and Arindam Banerjee (2010). "Generalized probabilistic matrix factorizations for collaborative filtering". In: *2010 IEEE International Conference on Data Mining*. IEEE, pp. 1025–1030.
- Shani, Guy, David Heckerman, and Ronen I Brafman (2005). "An MDP-based recommender system". In: *Journal of Machine Learning Research* 6.Sep, pp. 1265–1295.
- Singh, Ajit P and Geoffrey J Gordon (2008). "Relational learning via collective matrix factorization". In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 650–658.
- Sladic, Goran and Branko Milosavljević (June 2017). "Context-Aware Access Control for IoT Driven Processes". In:

- Strang, Thomas and Claudia Linnhoff-Popien (2003). "Service interoperability on context level in ubiquitous computing environments". In: *Intl. Conf. on Advances in Infrastructure for Electronic Business, Education, Science, Medicine, and Mobile Technologies on the Internet (SSGRR2003w)*. Citeseer.
- (2004). "A context modeling survey". In: *Workshop Proceedings*.
- Strang, Thomas, Claudia Linnhoff-Popien, and Korbinian Frank (2003). "CoOL: A context ontology language to enable contextual interoperability". In: *IFIP International Conference on Distributed Applications and Interoperable Systems*. Springer, pp. 236–247.
- Su, D and J Casamayor (2009). "Web-based Collaborative Working Environment and Sustainable Furniture Design". In: *Proceedings of the 19th CIRP Design Conference—Competitive Design*. Cranfield University Press.
- Su, Daizhong et al. (2005). "Collaborative design and manufacture supported by multiple Web/Internet techniques". In: *International Conference on Computer Supported Cooperative Work in Design*. Springer, pp. 483–492.
- Su, Xiaoyuan and Taghi M Khoshgoftaar (2009). "A survey of collaborative filtering techniques". In: *Advances in artificial intelligence 2009*.
- Suto, Hidetsugu and Patchanee Patitad (2015). "A representation model of collaboration in design process". In: *2015 10th Asian Control Conference (ASCC)*. IEEE, pp. 1–5.
- Tang, Jie et al. (2008). "ArnetMiner: Extraction and Mining of Academic Social Networks". In: *KDD'08*, pp. 990–998.
- Tarus, John K, Zhendong Niu, and Ghulam Mustafa (2018). "Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning". In: *Artificial intelligence review* 50.1, pp. 21–48.
- Taylor-Powell, Ellen and Boyd Rossing (2009). *Evaluating collaborations: Challenges and methods*.
- Tintarev, Nava and Judith Masthoff (2007). "A survey of explanations in recommender systems". In: *2007 IEEE 23rd international conference on data engineering workshop*. IEEE, pp. 801–810.
- Truong, Hong-Linh et al. (2008). "Incontext: A pervasive and collaborative working environment for emerging team forms". In: *2008 International Symposium on Applications and the Internet*. IEEE, pp. 118–125.
- Tversky, Amos (1977). "Features of similarity." In: *Psychological review* 84.4, p. 327.
- Van Rijsbergen, C (1979). "Information retrieval: theory and practice". In: *Proceedings of the Joint IBM/University of Newcastle upon Tyne Seminar on Data Base Systems*, pp. 1–14.
- Varelas, Giannis et al. (2005). "Semantic similarity methods in wordNet and their application to information retrieval on the web". In: *Proceedings of the 7th annual ACM international workshop on Web information and data management*, pp. 10–16.
- Wang, Ning (2016). "Towards a competency recommender system from collaborative traces". PhD thesis. Université de Technologie de Compiègne.
- Wang, Qiang et al. (2012). "Page recommendation based on user behavior in collaborative working environment". In: *Computer Engineering* 38.3, pp. 249–251.

- Wang, Xiaohang et al. (2004a). "Ontology Based Context Modeling and Reasoning using OWL." In: *Percom workshops*. Vol. 18. Citeseer, p. 22.
- Wang, Xiaohang et al. (2004b). "Semantic space: An infrastructure for smart spaces". In: *IEEE Pervasive computing* 3.3, pp. 32–39.
- Weiseth, Per Einar et al. (2006). "The wheel of collaboration tools: a typology for analysis within a holistic framework". In: *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*. ACM, pp. 239–248.
- Willmott, Cort J and Kenji Matsuura (2005). "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance". In: *Climate research* 30.1, pp. 79–82.
- Wilson, Paul (1991). *Computer supported cooperative work:: An introduction*. Springer Science & Business Media.
- Wouters, Laurent et al. (2017). "Collaborative systems engineering: Issues & challenges". In: *2017 IEEE 21st International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. IEEE, pp. 486–491.
- Xu, Jinsheng et al. (2008). "A survey of asynchronous collaboration tools". In: *Information Technology Journal* 7.8, pp. 1182–1187.
- Xu, Yunhong et al. (2010). "A Personalized Researcher Recommendation Approach in Academic Contexts: Combining Social Networks and Semantic Concepts Analysis." In: *PACIS*, p. 144.
- Yang, W. F., M. Wang, and Z. Chen (2014). "Fast Probabilistic Matrix Factorization for recommender system". In: *2014 IEEE International Conference on Mechatronics and Automation*, pp. 1889–1894. DOI: [10.1109/ICMA.2014.6885990](https://doi.org/10.1109/ICMA.2014.6885990).
- Ye, Juan et al. (2007). "Using situation lattices to model and reason about context". In: *Proceedings of MRC 2007*, pp. 1–12.
- Zhang, Xiaodan et al. (2008). "Medical document clustering using ontology-based term similarity measures". In: *International Journal of Data Warehousing and Mining (IJDWM)* 4.1, pp. 62–73.
- Zhang, Zijian, Lin Gong, and Jian Xie (2013). "Ontology-based collaborative filtering recommendation algorithm". In: *International Conference on Brain Inspired Cognitive Systems*. Springer, pp. 172–181.
- Zhu, Haozhe, Yanyan Shen, and Xian Zhou (2019). "TRPN: Matrix Factorization Meets Recurrent Neural Network for Temporal Rating Prediction". In: *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*. Springer, pp. 57–72.
- Zimmermann, Andreas, Andreas Lorenz, and Reinhard Oppermann (2007). "An operational definition of context". In: *International and Interdisciplinary Conference on Modeling and Using Context*. Springer, pp. 558–571.