

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Badji Mokhtar –Annaba-
Badji Mokhtar –Annaba- University



جامعة باجي مختار - عنابة-
Année: 2016

Faculté des sciences de l'ingénieur
Département d'informatique

THÈSE

Présentée en vue de l'obtention du diplôme de

Doctorat en Sciences

Intitulé :
Indexation de bases de données audio

Par
Nadia Lachtar

Devant le Jury

Présidente: Yamina Tlili

Directrice de thèse: Halima Bahi

Examineur: A/krim Amirat

Examineur: Karima Akli

Examineur: Zine eddine Bouras

Examineur: Smain Mazouzi

Pr. Université Badji Mokhtar Annaba

Pr. Université Badji Mokhtar Annaba

Pr. Université Med cherif Messaaidia Souk Ahras

MCA Université Houari Boumediene Alger

MCA Ecole Préparatoire aux Sciences et Technique Annaba

MCA Université 20 Août 1955 Skikda

Remerciements

Tous mes remerciements vont à :

Madame Bahi Halima, Professeur d'Informatique à l'Université Badji Mokhtar- Annaba, pour avoir accepté d'être ma directrice de thèse, pour l'intérêt qu'elle a accordé à ce travail, pour ses encouragements et son soutien,

Madame Tlili Yamina Professeur d'Informatique à Badji Mokhtar- Annaba, pour avoir accepté de présider mon jury de thèse,

Monsieur Amirat Abdelkrim Professeur d'Informatique à l'Université Souk-Ahras,

Madame Akli-Astouati Karima Maître de conférences à l'Université Houari Boumediene Alger,

Monsieur Zine Eddine Bouras Maître de Conférences à l'Ecole Préparatoire Aux sciences et Techniques Annaba et

Monsieur Smaine Mazouzi Maître de conférences à l'Université 20 Août 1955-Skikda,

Pour avoir accepté d'examiner cette thèse et pour l'intérêt qu'ils ont porté à ce travail.

Mes derniers remerciements qui ne sont pas les moindres sont pour les personnes les plus chères dans ma vie, mes parents qui ont tout sacrifié pour moi, mon mari qui m'a toujours soutenu dans tout ce que j'entreprenais, mes enfants, mon frère et mes sœurs. Ce travail est le fruit de leurs patiences et leurs sacrifices.



Résumé

Aujourd'hui, les ressources musicales disponibles sur le web augmentent exponentiellement. Dans cet énorme entrepôt de données musicales, les systèmes d'indexation et de recherche d'information musicale actuels ne permettent pas de retourner aux utilisateurs les chansons répondant exactement à leurs besoins exprimés par une requête sur une collection de chansons à cause des techniques d'indexation utilisées. Pour améliorer la pertinence de la recherche d'information musicale, nous proposons dans cette thèse premièrement une approche qui se base sur l'utilisation d'un algorithme de colonie de fourmis pour la catégorisation de chansons puis pour permettre la prise en compte des problèmes d'homonymie, de synonymie et de contextualisation liés à l'emploi du langage naturel par les utilisateurs. Nous proposons deuxièmement l'utilisation d'une ontologie de domaine pour l'indexation d'une base de chansons et l'utilisation des liens sémantiques entre chansons pour permettre l'inférence de toutes les chansons pertinentes. En plus nous proposons un algorithme de raffinement de requêtes pour la recherche de chansons.

Cette approche est testée et validée sur le domaine de chansons Algérienne dans le contexte du web sémantique.

Une application web est développée afin d'indexer les chansons par des concepts et effectuer une recherche sémantique de chansons et exploiter les liens concepts/termes et les relations sémantiques structurant l'ontologie.

Mots clés: Catégorisation de chansons, Recherche d'information musicale, Ontologie, Web sémantique, Musique algérienne, Swarm intelligence, Algorithme de colonie de fourmis.

Abstract

Nowadays, the musical resources available on the web increase exponentially. In this huge repository of musical data, current music information search and indexing systems do not allow users to return the songs that correspond to their needs expressed by a query due to the used indexing techniques.

To improve the relevance of the music information search, we propose in this thesis firstly an approach based on the use of an ant colony algorithm for songs categorization and then to allow consideration of the problems of homonymy, synonymy and contextualization related to the use of natural language by users. We suggest, secondly the use of domain ontology to index a collection of songs and the use of semantic links in order to allow the inference of all relevant songs. In addition, we propose a query refinement algorithm to find songs.

This approach is tested and validated in the domain of Algerian songs in the context of the semantic web. A web application is developed to index the songs by concepts, perform a conceptual search of songs and exploit concepts / terms links and semantic relationships that structure the ontology.

Keywords: Song categorization, Music search information, Ontology, Semantic web, Algerian music, Swarm intelligence, Ant colony algorithm.

ملخص

حالياً، الموارد الموسيقية المتاحة على شبكة الإنترنت تتزايد بسرعة كبيرة كما أن نظم الفهرسة والإسترجاع للمعلومات الموسيقية الحالية في هذا المستودع الهائل من البيانات الموسيقية، لا تتيح للمستخدمين إيجاد الأغاني التي تلبى احتياجاتهم بالضبط وهذا بسبب تقنيات الفهرسة المستخدمة. بغية تحسين البحث عن المعلومات الموسيقية، نقترح في هذه الأطروحة أولاً اتباع نهج يقوم على استخدام خوارزمية مستعمرة النمل لتصنيف الأغاني ثم من أجل الأخذ بعين الإعتبار المشاكل المتعلقة باستخدام اللغة الطبيعية من طرف المستخدمين نقترح، استخدام أنطولوجيا المجال لفهرسة قاعدة بيانات من الأغاني واستخدام الوصلات الدلالية بين الأغاني لإيجاد جميع الأغاني بالإضافة إلى ذلك نقترح خوارزمية للبحث عن الأغاني.

تم اختبار هذا الأسلوب والتحقق من صحته على الأغاني الجزائرية حيث تم تطوير "تطبيق ويب" لتصنيف الأغاني حسب المفاهيم وأداء بحث دلالي واستغلال روابط المفاهيم والمصطلحات والعلاقات الدلالية المهيكلة للأنطولوجيا.

كلمات مرشدة : تصنيف الأغاني، البحث عن المعلومات الموسيقية، أنطولوجيا، الويب الدلالي، الموسيقى الجزائرية، الذكاء السربي، خوارزمية مستعمرة النمل

Table des matières

Introduction Générale.....	13
Problématique de la recherche.....	13
Domaine de recherche	14
Plan de la thèse	14
Chapitre I : Indexation audio et spécificité de l'information musicale	16
I. Indexation audio	18
I.1. Introduction	18
I.2. L'indexation audio	19
I.2.1. Définition de l'indexation audio.....	19
I.2.2. Aspects de l'indexation audio	19
I.2.2.1. La segmentation parole/musique	19
I.2.2.2. L'identification de langue	20
I.2.2.3. Le suivi de locuteur.....	20
I.2.2.4. L'indexation par locuteur.....	21
I.2.2.5. La détection de mots-clés	22
I.3. L'identification du thème	22
I.3.1. Analyse sémantique latente (Latent Semantic Analysis LSA).....	23
I.3.2. Ontologie	24
I.3.3. Identification thématique hiérarchique	25
I.4. Conclusion.....	26
II. L'information musicale	27
II.1. La recherche de l'information musicale	27
II.2. Typologie des documents musicaux	27
II.3. Les facettes de l'information musicale	28
II.4. Problèmes de l'indexation d'information Musicale.....	30
II.4.1. Les représentations multiples de l'information musicale et sa complexité.....	31
II.4.2. La complexité des requêtes musicales	32
II.4.3. L'évaluation des résultats d'une recherche d'information musicale.....	33
II.5. Méthode et outils de recherche d'information Musicale	34

II.5.1. Méthodes et offre d'outils disponibles	34
II.5.2. Exemples de méthodes et d'outils.....	34
II.6. Revue de la littérature	35
II.7. Conclusion	37
Chapitre II: Swarm Intelligence et colonies de fourmis pour la catégorisation de chansons.....	38
I. Catégorisation de chansons.....	41
I.1. Introduction	41
I.2. Catégorisation de chansons selon le texte	41
I.2.1. Définition de la catégorisation.....	41
I.2.2. Comment catégoriser une chanson?	42
I.2.3. Représentation et codage d'une chanson.....	42
I.3. Catégorisation de chansons en utilisant l'algorithme Bayes naïf.....	44
I.4. Critères de mesure de performances des classificateurs	46
I.4.1. Le rappel	46
I.4.2. La précision	47
I.4.3. F-mesure	47
I.5. Conclusion.....	47
II. Swarm Intelligence et colonies de fourmis	48
II.1. Introduction.....	48
II.2. Intelligence collective dans la nature.....	48
II.3. Particularités du comportement collectif	49
II.4. Introduction à l'intelligence en essaim (swarm intelligence SI).....	49
II.5. Modèles de l'intelligence en essaim	51
II.6. Principes de l'intelligence en essaim	51
II.7. Essaims et vie artificielle	53
II.8. Techniques de la Swarm Intelligence	53
II.8.1. L'optimisation par colonies de fourmis (ACO)	53
II.8.2. L'optimisation par colonies d'abeilles (BCO)	55
II.8.3. L'optimisation par essaim de particules (PSO).....	56
II.9. Applications de la swarm intelligence	57
II.10. Avantages de la swarm intelligence.....	57

II.11. L'approche d'optimisation par colonies de fourmis	58
II.11.1. Inspiration biologique	58
II.11.2. Théorie de stigmergie.....	59
II.11.3. Auto-organisation et Rétroaction	59
II.11.4. Structure d'ACO	59
II.11.5. Métaheuristique ACO	60
II.11.6. Algorithmes d'optimisation par colonies de fourmis.....	61
II.11.7. Exemple d'utilisation d'ACO dans un problème de classification de textes	65
II.12. Conclusion	67
III. Application d'un algorithme de colonies de fourmis à la catégorisation de chansons	68
III.1. Introduction	68
III.2. Ressemblance de la catégorisation de chanson avec le problème de voyageur de commerce (TSP).....	68
III.3. Le processus d'apprentissage	69
III.4. Construction du graphe de chansons	70
III.5. Calcul de distances entre la chanson à classer et les chansons constituant le graphe	71
III.6. Construction d'une solution	72
III.7. Description de l'algorithme.....	73
III.8. Expérimentations et discussions.....	76
III.8.1. Résultats.....	76
III.8.2. Le temps d'exécution.....	79
III.9. Conclusion.....	81
Chapitre III: L'indexation et la recherche conceptuelle de chansons	82
I. Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies.....	84
I.1. Web sémantique	84
I.1.1. Définition.....	84
I.1.2. Principe général	85
I.2. Les ontologies pour le web sémantique	85
I.2.1. Définition d'une ontologie	85
I.2.2. Les ontologies et la représentation des connaissances	85

I.2.3. Les constituants d'une ontologie	86
I.2.4. Les langages d'ontologies	89
I.2.5. Les outils de construction d'ontologies	90
I.3. Construction de l'ontologie de chansons	91
I.3.1. Liste des concepts de l'ontologie de domaine	92
I.3.2. Liste des attributs.....	94
I.3.3. Liste des relations	95
I.3.4. Diagramme de classes de l'ontologie de domaine pour la description des chansons	97
I.4. Création de l'ontologie de domaine de chansons.....	99
I.4.1. Création des classes de l'ontologie de domaine de chansons.....	100
I.4.2. Création des propriétés (les attributs) de l'ontologie de domaine de chansons	101
I.4.3. Création des relations de l'ontologie de domaine de chansons	102
I.5. Conclusion.....	103
II. La catégorisation conceptuelle multi-langues de chansons.....	104
II.1. Introduction.....	104
II.2. Conception générale du système d'indexation et de catégorisation de chansons.....	104
II.2.1. Lexicalisation	104
II.2.2. Classification.....	105
II.2.2.1. Vecteur de caractéristiques	105
II.2.2.2. Désambiguïsation	105
II.2.2.3. Classification	107
II.3. Expérimentations	108
II.4. Conclusion	111
III. La recherche conceptuelle de chansons	112
III.1. Introduction	112
III.2. L'indexation et la recherche conceptuelle.....	112
III.2.1. Conception générale du système d'indexation et de recherche de chansons.....	112
III.2.1.1. Ontologie de domaine	112
III.2.1.2. Module Analyse	113
III.2.1.3. Extracteur de triplets	113
III.2.1.4. Module de recherche	114
III.3. Implémentation de l'indexation et la recherche conceptuelle	116

III.3.1. La recherche conceptuelle	116
III.3.2. Approche d'indexation et de recherche par mots-clés.....	118
III.4. Evaluation de l'approche	120
III.4.1. Mesures d'évaluation.....	120
III.4.2. Discussion.....	122
III.4.3. Exemple d'illustration.....	122
III.4.3.1. Exemple Recherche conceptuelle en remplissant le formulaire.....	122
III.4.3.2. Exemple Recherche conceptuelle en remplissant la requête en langage naturel .	123
III.5. Conclusion.....	125
Conclusion générale	126
Perspectives.....	128
Publications	129
Références bibliographiques	130

Table des Figures

Figure 1: Quelques descripteurs qui peuvent être extraits d'un document audio	18
Figure 2: Les sept facettes de l'information musicale	29
Figure 3: Illustration de l'expérience réelle du pont à double branche	58
Figure 4: Des fourmis suivent une piste de phéromone	59
Figure 5: Méta-heuristique ACO	60
Figure 6: Représentation du graphe de catégorisation de chansons	70
Figure 7: Déplacement d'une fourmi	73
Figure 8: Histogramme montrant le taux de réussite pour chaque catégorie pour les 2 algorithmes	79
Figure 9: Relation entre le temps d'exécution et le nombre de chansons	80
Figure 10: Les couches du standard du Web sémantique	84
Figure 11: Diagramme de classes de l'ontologie de domaine	98
Figure 12: Création des classes de l'ontologie de chansons	100
Figure 13: Création des propriétés de l'ontologie de chansons	101
Figure 14: Création des relations de l'ontologie de chansons	102
Figure 15: Vecteur de caractéristiques 1.....	107
Figure 16: Vecteur de caractéristiques 2.....	107
Figure 17: Rappel de la classification	110
Figure 18: La précision de la classification	110
Figure 19: Histogramme montrant la F_1 mesure pour chaque catégorie pour les 2 algorithmes	111
Figure 20: Conception du Module d'indexation et de Recherche	112
Figure 21: Architecture logicielle du système développé	116
Figure 22: Interface de l'application de recherche web (a) : recherche simple, (b) recherche avancée, (c) indexation et insertion d'une nouvelle chanson	117
Figure 23: Interface de l'application de recherche conceptuelle Web	118
Figure 24: Résultats de la recherche pour la chanson "One Two Three Viva l'Algérie	119
Figure 25: Ajout de chansons	119
Figure 26: Résultat de recherche de la chanson one two three viva l'Algérie et écoute de la chanson	120
Figure 27: La courbe rappel-précision	122
Figure 28: Exemple de recherche de la chanson Aicha	123
Figure 29: Réponse du système pour la recherche de la chanson Aicha	123
Figure 30: Recherche d'une requête exprimée en langage naturel	124

Liste des tableaux

Tableau 1 : Classes des corpus	76
Tableau 2: Résultats des tests avec l’algorithme des colonies de fourmis	77
Tableau 3: Resultats des tests avec l’algorithme Bayes naïf	77
Tableau 4: Rappel, Précision et F1 pour chaque classe (colonie de fourmis).....	78
Tableau 5 : Rappel, Précision et F1 pour chaque classe (Bayes naïf)	78
Tableau 6: Relation entre le temps d’exécution et le nombre de chansons	80
Tableau 7: Liste des concepts	92
Tableau 8: Liste des attributs	94
Tableau 9 : Liste des relations	95
Tableau 10 : Classes des corpus	109
Tableau 11: Matrice de confusion avec classification conceptuelle	109
Tableau 12: Matrice de confusion avec l’algorithme Bayes naïf	109
Tableau 13: Précisions aux in points de rappels pour les deux méthodes	121
Tableau 14 : Les triplets de la requête	124

Introduction Générale

Problématique de la recherche

A l'heure actuelle, nous disposons d'une quantité énorme d'information audio à la fois importante et grandissante par le biais des bases de données publiques ou privées (sites internet, Cédéroms,...), mais sans aucune maîtrise de contenu. La description par mots-clés est peu adaptée à la richesse de cette information, puisqu'elle entraîne une indexation subjective et coûteuse (à cause de l'importante intervention humaine). Avec le développement de l'internet, beaucoup de données audio ne sont pas actuellement indexés par les moteurs de recherches web. L'indexation audio consiste à trouver de bons descripteurs audio qui vont être utilisés comme indices pour l'archivage et la recherche. Le domaine de l'indexation audio tente donc de répondre au besoin d'outils automatiques de description de contenu audio afin d'améliorer l'accès. Une grande portion d'information audio est disponible sur le Web, le résultat est que l'utilisateur du web gaspille une grande partie de son temps à examiner un grand nombre de page Web en cherchant ce qui lui convient, car le Web ne fournit pas un service dans ce sens. Il est possible d'indexer l'information du point de vue syntaxique, ce que font la majorité des moteurs de recherche, mais les choses se corsent lorsqu'on cherche à obtenir la structure réelle du savoir contenu dans un document audio. Devant l'abondance de l'offre, il devient difficile pour les usagers de s'y retrouver. Il est donc essentiel de trouver de nouvelles façons de chercher, conserver, mettre à jour, analyser les données accessibles, indexer et classifier les documents audio. Le web sémantique est donc la solution, cette nouvelle vision du web consiste à rendre les ressources web non seulement compréhensible par les humains mais également par les machines.

L'objectif de ce travail de thèse est de proposer des approches d'indexation de chansons qui permettent la classification de chansons selon un ensemble de catégories prédéfinies. Pour atteindre cet objectif, nous avons choisi d'utiliser des algorithmes d'apprentissage supervisé de classification de chansons basés sur les métadonnées et les paroles et nous avons proposé premièrement l'utilisation d'un algorithme de colonie de fourmis pour la classification des chansons en utilisant le titre, les refrains et les caractéristiques de l'artiste ensuite, nous avons créé une ontologie de domaine pour représenter et indexer les chansons en utilisant les relations et les liens sémantiques et nous avons développé un algorithme pour la recherche conceptuelle de chansons. L'utilisation des colonies de fourmis nous permet d'économiser le temps et l'effort nécessaires pour classer une chanson. Toutefois, reste un problème de représentation de chansons à

l'aide de moyens qui favorisent la réutilisation. En plus le traitement automatique de l'indexation utilisant les colonies de fourmis ne fonctionne que sur des principes statistiques, il ne permet pas de prendre en compte les problèmes d'homonymie, de synonymie et de contextualisation liés à l'emploi du langage naturel par les utilisateurs lors de la recherche de chansons. Dans cette perspective, nous avons proposé d'explorer la méthode d'indexation en utilisant une ontologie de chansons. Un outil de repérage et d'exploration de chansons sur le Web. Notre approche modélise et indexe des chansons en vue de leur catégorisation en fonction de leurs thèmes. En effet, il est fréquent que les demandes sur le web soient liées à un thème pas à une chanson.

Domaine de recherche

Le thème de cette thèse s'inscrit dans le domaine de l'indexation de base de données audio et s'intéresse aux fichiers musicaux à l'échelle du Web et se situe à l'intersection du web sémantique avec le domaine de la recherche musicale (Music Information Retrieval MIR). Dans ce vaste domaine de recherche qui suscite un grand intérêt on s'intéresse à l'indexation de chansons en utilisant les métadonnées et les paroles pour identifier leurs thèmes. Nous visons à modéliser les chansons en créant une ontologie et à développer une application web afin d'indexer et d'identifier le thème d'une chanson et l'insérer dans l'ontologie.

Plan de la thèse

Outre l'introduction, Cette thèse est composée de trois chapitres :

Le premier chapitre introduit le contexte dans lequel se place cette thèse en discutant certains concepts fondamentaux liés à l'aspect de l'indexation audio en général, et spécifiquement à l'information musicale. Il dresse un état de l'art sur les différentes approches qui sont en relation avec le contexte scientifique dans lequel se déroule cette thèse. Les deux chapitres suivants présentent notre approche pour construire des systèmes de catégorisation de chansons en s'inspirant de la nature (colonie de fourmis), puis en utilisant des outils des ingénieries du web sémantique pour construire une ontologie de domaine et effectuer une indexation et une recherche conceptuelle de chansons. Ils décrivent leurs mises en œuvre et leurs évaluations.

Plus précisément :

Introduction Générale

Le chapitre 1 « Indexation audio et spécificité de l'information musicale » est composé de deux parties. La première partie présente certains aspects de l'indexation audio et l'identification de thèmes. Quand à la deuxième partie elle est consacrée à la recherche d'information musicale, à la typologie des documents musicaux, ainsi qu'aux diverses facettes que revêt l'information musicale. Aux représentations multiples de l'information musicale et sa complexité. Aux problèmes de l'indexation de la musique et aux diverses méthodes de sa recherche. Enfin, elle dresse un état de l'art des travaux utilisant les paroles pour la catégorisation et la recherche des chansons.

Le chapitre 2 « Swarm intelligence et colonies de fourmis pour la catégorisation de chansons » est aussi composé de trois parties. La première partie expose la représentation de chansons, utilise l'algorithme Bayes naïf pour catégoriser une chanson et en fin, elle présente les critères de mesure de performances. La seconde partie porte sur la swarm intelligence en présentant les concepts de bases de l'intelligence en essaim ainsi que les modèles et les techniques qui lui sont associées. Elle se concentre particulièrement sur les colonies de fourmis. Elle présente les différents concepts liés à l'optimisation par colonies de fourmis ainsi que l'algorithme ACO associé à cette approche et son utilisation en Data Mining et, plus précisément, pour la classification de textes. La troisième partie présente notre contribution au problème posé dans cette thèse, à savoir la proposition d'une approche de catégorisation de chansons qui consiste à adapter la métaheuristique ACO pour la classification de chansons. Elle se termine par une évaluation et présente les résultats obtenus.

Le chapitre 3 « L'indexation et la recherche conceptuelle de chansons » est divisé en trois parties. La première partie décrit l'apport du Web sémantique, la notion d'ontologie et les outils pour la création d'ontologies. Elle détaille la conception et la construction de l'ontologie de domaine pour la représentation des chansons. En fin, elle utilise l'outil Protégé pour la création et l'édition de l'ontologie. La seconde partie présente de manière exhaustive le cœur de notre travail de thèse, elle utilise l'ontologie de domaine, les relations et les liens sémantiques pour classer une chanson dans un environnement multi-langues. L'approche est évaluée sur les chansons algériennes et comparée à la catégorisation en utilisant l'algorithme Bayes naïf. La troisième partie aborde la recherche conceptuelle de chansons et expose un algorithme de raffinement de requêtes. Une application web est développée. Des expériences sont décrites et des résultats et discussions sont exposés à la fin. Finalement, la thèse se termine par une conclusion, souligne une synthèse des contributions et introduit une mise en perspective de nos travaux par rapport aux résultats obtenus.

Chapitre I : Indexation audio et spécificité de l'information musicale

Chapitre I

Ce chapitre est divisé en deux parties:

La première partie définit l'indexation audio et discute certains aspects de l'indexation qui peut consister en: la segmentation de la parole et de la musique, l'identification de langue, le suivi de locuteur, l'indexation de locuteur, l'indexation de mots-clés ou l'identification de thèmes. Elle se termine par une conclusion.

La seconde partie est consacrée à la recherche d'information musicale, à la typologie des documents musicaux, ainsi qu'aux diverses facettes que revêt l'information musicale. Aux représentations multiples de l'information musicale et sa complexité. Elle aborde les problèmes de l'indexation de la musique et aux diverses méthodes de sa recherche.

Enfin, elle dresse une revue de la littérature des travaux utilisant les paroles pour la catégorisation et la recherche des chansons.

I. Indexation audio

I.1. Introduction

Beaucoup de travaux ont réussi à indexer, à rechercher et à parcourir un signal de type audio en utilisant les dernières avancées du domaine de reconnaissance automatique de la parole [1, 2, 3].

Les applications possibles dans ce domaine sont nombreuses, nous pouvons citer l'indexation de bases de données selon des mots-clés connus lors des requêtes, ou une indexation par locuteur, par thème ou par langue.

Dans ce qui suit, nous allons d'abord décrire certaines méthodes utilisées pour structurer et indexer des documents audio.

A partir d'un document audio, on peut extraire un grand nombre d'informations non-verbales qui peuvent être très structurantes et peuvent conduire à l'extraction de descripteurs pertinents, La figure 1 montre qu'on peut séparer des segments de paroles des segments de musiques, détecter les sons clés (comme les jingles), identifier la langue d'un segment, suivre un locuteur connu, détecter certains mots-clés ou extraire les thèmes principaux.

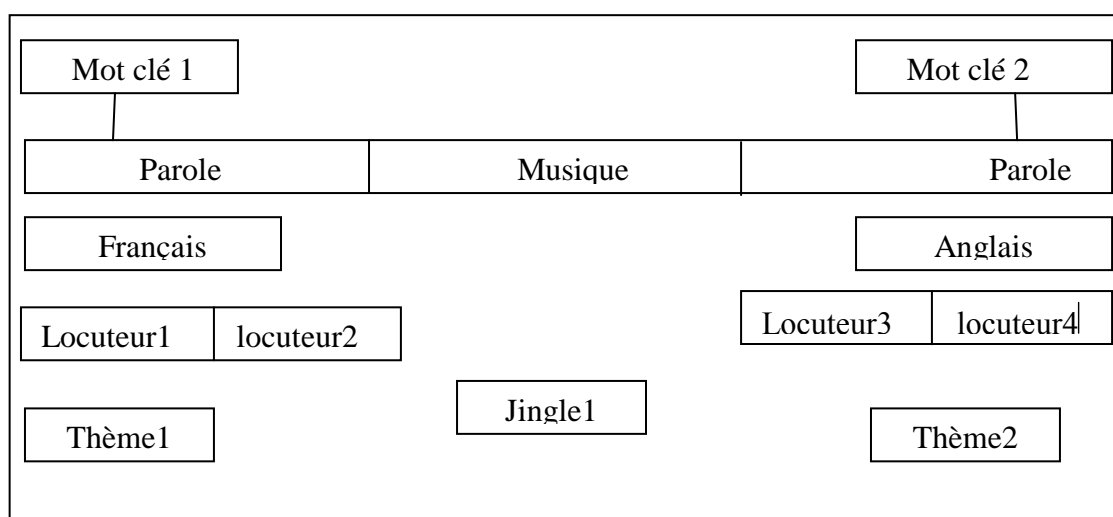


Figure 1: Quelques descripteurs qui peuvent être extraits d'un document audio (d'après [11])

I.2. L'indexation audio

I.2.1. Définition de l'indexation audio

L'AFNOR (Association Française de NORmalisation) définit l'indexation comme une représentation de résultats d'une analyse en langue naturelle ou dans un langage normalisé d'un document. Une définition plus classique, non contradictoire, est l'identification et la localisation de séquences pertinentes ou de thèmes majeurs au sein d'un document par une analyse de son contenu. L'objectif est, à l'aide de ses index, de classer ultérieurement le document audio parmi un ensemble de documents audio d'une collection donnée, d'extraire le contexte de cet index au sein du document audio lui-même. Ce type d'indexation a pour but l'optimisation de l'accès aux données dans de grandes bases.

À l'heure actuelle, les méthodes d'indexation en audio sont principalement manuelles: un opérateur humain doit écouter le document numérique de façon à sélectionner les informations recherchées. Mais, compte tenu de l'accroissement gigantesque du volume de données à traiter, la tâche d'indexation devient extrêmement fastidieuse, et l'automatisation semble désormais indispensable. Afin de synthétiser l'information pertinente, naviguer ou rechercher efficacement dans les bases de données audio, des systèmes d'indexation doivent être développés pour pouvoir exploiter ces nouvelles technologies numériques.

I.2.2. Aspects de l'indexation audio

L'indexation de bases de données audio revêt plusieurs aspects, selon l'utilisation que nous souhaitons pour les données.

I.2.2.1. La segmentation parole/musique

Le but de cette approche est de délimiter les segments de parole et de musique dans le document afin de le segmenter en des portions parole et des portions musique. La plupart des systèmes de transcription des journaux télévisés utilisent ce type de séparation des segments de musique de la parole afin de limiter l'application du système de transcription de la parole aux segments de parole [1]. Cette segmentation est souvent liée à une structuration spécifique du document (publicités, jingles etc...).

Partie I : Indexation audio

Généralement, la segmentation de la parole et la musique est basée sur les différences acoustiques entre les deux types de sons. Dans [2, 3, 4, 5] la discrimination est basée sur des ensembles d'indices tels que la moyenne et la variance des taux de passage par zéro, le spectre, etc... Toutes ces caractéristiques ont généralement des propriétés de discrimination intéressantes lorsqu'elles prennent des valeurs très différentes pour chaque catégorie de sons.

Différents classificateurs sont couramment utilisés: les modèles de mélange de gaussiennes, les K-plus proches voisins, les réseaux de neurones. Dans [6,7], une paramétrisation cepstrale classique associée avec le modèle de mélange de gaussiennes donnent de bons résultats. Ce qui semble important dans la séparation de la parole et de la musique est la notion d'indépendance entre les deux indices de suivi. La modélisation différenciée permet d'exploiter les différences structurelles entre la parole et la musique. Chaque classe a son propre ensemble de paramètres (cepstrale pour la parole, des filtres pour la musique) et des modèles (un modèle de mélange de gaussiennes de 32 composantes pour la parole et un modèle de mélange de gaussiennes de 10 composantes pour la musique).

I.2.2.2. L'identification de langue

L'identification de langue consiste à identifier le langage d'un document audio ou d'un segment dans un document audio. Cette tâche est apparue récemment dans le cadre de la reconnaissance de la parole multilingue, mais n'a pas été utilisée dans l'indexation audio. Dans l'état de l'art, on trouve une approche qui utilise une analyse acoustique à base de cepstre et plusieurs systèmes de reconnaissance de phone dépendants du langage suivie par une modélisation n-gram spécifique pour chaque langage à identifier [8]. Cette approche exige que les données d'apprentissage étiquetées soient disponibles pour plusieurs langues, mais pas nécessairement pour tous.

Parmi les autres approches qui ont été proposées, certaines recherches travaillent au niveau de l'analyse acoustique [9, 10] et d'autres se proposent d'explorer divers indices pour identifier divers langages tel que le rythme [11] par exemple.

I.2.2.3. Le suivi de locuteur

Le suivi de locuteur consiste à estimer un début et un temps de fin de chaque segment dans lequel un locuteur cible parle. Cette tâche est généralement abordée par une approche statistique. Pour détecter un modèle de locuteur cible dans un document audio deux méthodes sont construites: un

Partie I : Indexation audio

modèle de locuteur cible et un modèle de base qui vise à représenter la parole des locuteurs autre que le locuteur cible ou d'autres types de sons. Ces modèles sont construits en utilisant les caractéristiques des vecteurs extraites des segments étiquetés. Dans le cas de plusieurs locuteurs cibles, au moins un modèle est nécessaire pour chaque locuteur cible. Enfin, plusieurs modèles de base représentant le son des différents types de locuteurs non cible et plusieurs modèles pour chaque locuteur cible, représentant les différents qualités de parole. Cette approche est étudiée dans [10,11].

I.2.2.4. L'indexation par locuteur

L'indexation par locuteurs d'une base audio tient une place essentielle, il s'agit de savoir qui parle et quand afin de saisir la cohérence du dialogue. Elle consiste à organiser les données en fonction des locuteurs présents dans la base de données. Elle est composée de plusieurs étapes: la segmentation de chaque document audio en locuteurs, le regroupement des segments de locuteurs, la modélisation des locuteurs et en fin, la reconnaissance de la séquence de locuteurs. La segmentation en locuteurs vise à obtenir des segments ne contenant les paroles que d'un seul locuteur et les plus longs possibles. Une technique de segmentation est mise en œuvre en deux phases, la première étape détecte les changements de locuteurs potentiels. Elle repose sur le calcul du rapport de vraisemblance généralisé entre deux portions de signal. La seconde étape de la technique de segmentation est basée sur le critère d'information Bayésien qui permet de valider ou non les points de changements de locuteurs obtenus, l'étape suivante consiste à regrouper tous les segments appartenant à un seul locuteur, afin d'obtenir un important volume de données du dit locuteur pour la construction d'un modèle fiable. Le rapport de vraisemblance et le critère d'information Bayésien ayant prouvé leurs efficacités au cours de la segmentation seront utilisés respectivement comme critère de regroupement et comme critère d'arrêt pour le regroupement hiérarchique.

Le problème de la segmentation par locuteurs est généralement adressé par l'une des deux méthodes suivante: la première décrite dans [11], repose sur la détection de changement des locuteurs suivie d'une étape de regroupement, la seconde décrite aussi dans [11], fait la segmentation et le regroupement en même temps en utilisant les modèle de Markov cachés. Dans les deux cas, le système d'indexation doit déterminer l'ensemble des locuteurs présents dans un document de données audio, et doit répondre aux hypothèses suivantes: aucune connaissance a

priori sur les locuteurs ou sur le langage, le nombre de locuteurs est inconnu et les personnes ne parlent pas simultanément. Les modèles énoncés des locuteurs sont construits durant le processus de segmentation. Le lien des énoncés des locuteurs [11] est un problème de classification similaire au regroupement de locuteurs.

I.2.2.5. La détection de mots-clés

Les systèmes de diffusion de journaux télévisés sont basés sur une transcription complète des segments de parole tandis que les applications de recherche d'information sur le web préfèrent la détection de mots-clés qui consiste à détecter un ensemble plus au moins important de mots-clés à partir d'un flux de parole. Les systèmes de détection de mots-clés sont généralement basés sur les modèles de Markov cachés et sont considérés comme plus efficaces pour la modélisation arbitraire de la parole que les systèmes à base de patron [13].

Deux approches principales se trouvent dans la littérature, la plus évidente consiste à utiliser un système de reconnaissance de parole continu avec un large vocabulaire pour produire une chaîne de mots. Ensuite, les algorithmes de recherches sont appliqués pour la détection des mots-clés dans cette chaîne [13]. Cette approche donne les meilleurs résultats [13]. L'autre approche est fondée sur l'utilisation de modèles de mots-clés et de modèles poubelles. Les modèles poubelles peuvent être des modèles sous mots tels que des modèles phonétiques ou peuvent être des modèles tous mots [13]. Le rôle des modèles poubelles est d'absorber les mots hors-vocabulaire dans le flux de paroles.

I.3. L'identification du thème

L'auteur dans [12], définit un thème comme un événement majeur. En ce qui concerne la parole, un thème est défini par un ensemble de mots pertinents [12]. Ceci implique l'utilisation d'un moteur de reconnaissance automatique de la parole comme une des approches frontales récentes estimant que les méthodes à base de phonèmes sont plus précises.

L'auteur dans [12], considère trois tâches différentes: la segmentation du récit, la détection de thèmes et le suivi de thème. La segmentation du récit est la tâche qui consiste à segmenter le flux de la parole en histoire de thèmes cohérents. La tâche de suivi consiste à associer de nouvelles histoires avec des thèmes connus. Tandis que la détection des thèmes est la tâche de détection et

suivi des thèmes non connus auparavant par le système. Ces trois tâches peuvent être intégrées dans un système d'indexation automatique de la parole.

Indexer un document consiste à lui donner une représentation qui permet de faciliter l'accès à l'information (contenu du document). L'opération d'indexation est particulièrement difficile dans la mesure où elle pose le problème de la représentation du sens. Beaucoup de travaux ont traité l'amélioration de l'indexation des documents audio à l'aide de l'analyse sémantique latente et les cartes auto-organisatrices de Kohonen [12], et à l'aide des ontologies aussi [14]. D'autres travaux proposent des modèles statistiques à base d'unigramme pour identifier automatiquement le thème d'un document parmi une arborescence prédéfinie de thèmes possibles [15].

I.3.1. Analyse sémantique latente (Latent Semantic Analysis LSA)

L'**analyse sémantique latente** (Latent semantic analysis): est un procédé de traitement des langues naturelles. Elle permet d'établir des relations entre un ensemble de documents et les termes qu'ils contiennent, en construisant des concepts liés aux documents et aux termes. La LSA utilise une matrice qui décrit l'occurrence de certains termes dans les documents. Les lignes correspondent aux termes et les colonnes correspondent aux documents. Les termes sont généralement des mots tronqués ou ramenés à leur radical, issus de l'ensemble du corpus. On a donc le nombre d'apparition d'un mot dans chaque document, et pour tous les mots. Ce nombre est normalisé en utilisant la pondération $tf-idf$ (term frequency – inverse document frequency).

Dans [12], les méthodes d'indexations utilisant la LSA sont évaluées pour deux bases de données de diffusion de nouvelles (l'une française et l'autre anglaise), en utilisant la moyenne de perplexité des documents et les requêtes d'essais analysées par un expert.

Lorsqu'un document n'est pas trouvé alors qu'il répond à une demande, dans ce cas, les mots de l'interrogation ne sont pas identiques à ceux du texte, mais ils sont des synonymes. Dans le cas contraire, les homonymes provoquent le rappel de textes non pertinents. Les textes sont porteurs d'une structure sémantique implicite dont on tente d'extraire les concepts en tant qu'unité de sens.

I.3.2. Ontologie

Une ontologie du domaine dont traite le document permet d'associer à ce dernier un réseau d'informations valable non seulement pour le document en question, mais aussi pour tous ceux indexés conformément à la même ontologie. L'interopérabilité et la réutilisabilité sont à l'évidence des retombées importantes pour des documents indexés sur la base d'une même ontologie. Au-delà de la rigueur lexicale, l'approche ontologique présente plusieurs autres avantages. L'ontologie fixe un certain nombre de règles sémantiques générales d'hyponymie (relation 'est une sous classe de'), d'hyperonymie (relation 'est une super classe de'), de méréonymie (relation 'se compose de'), et de relations spécifiques au domaine (causalité, chronologie, etc.). La disponibilité de ces règles joue le rôle de contraintes qui favorisent un certain nombre de qualités. En effet, indexer un document constitue une tâche complexe, fastidieuse, très coûteuse et au résultat très subjective. Le support d'une ontologie permet en partie de prendre en compte ces difficultés. Avec une ontologie à sa disposition, l'indexeur (la personne ou l'outil qui indexe) peut se reposer sur une connaissance formalisée du domaine cible. Sa tâche se résume à instancier des concepts et des relations sans avoir à reconstruire un corpus de connaissances sur le domaine. Un autre avantage concerne la navigation et la recherche. Plus généralement, une ontologie permet d'étendre et renforcer la calculabilité en matière de similarité sémantique. Des agents intelligents peuvent utiliser les relations proposées par l'ontologie pour effectuer des inférences conceptuelles afin de ne pas limiter les capacités des moteurs à la seule lecture des indexations disponibles à l'intérieur d'un document.

L'ontologie permet aussi de favoriser l'évaluation d'un document et de l'indexation associée. Un moteur peut calculer les trous conceptuels, le niveau de redondance, le niveau de généralité d'un document. Ces calculs sont liés à l'analyse de couverture ontologique i.e. aux nombres et à la nature des concepts de l'ontologie instanciés dans le document.

Cependant l'utilisation d'une ontologie pour indexer un document présente aussi certains inconvénients. La première difficulté est évidente: il faut disposer d'une ontologie. Deux solutions sont possibles.

La première consiste à réutiliser une ontologie disponible en ligne mais cela entraîne plusieurs inconvénients [15]. Il faut se plier au formalisme imposé par l'ontologie en ligne comme par exemple Ontolingua, ou KIF, et adapter ses outils à ces formalismes. Si la standardisation est ainsi favorisée, la conception se complique quand le but recherché est de développer un environnement

d'indexation à usage ciblé. De plus les ontologies importées sont souvent trop générales et trop vastes. L'information utile est noyée dans une masse d'informations annexes.

La deuxième solution consiste à construire sa propre ontologie. Il est alors possible de cibler l'information ontologique, mais les qualités d'économie et de réutilisabilité ne sont plus respectées puisque l'ontologie et sa représentation sont spécifiques. Par ailleurs, les ressources et compétences humaines nécessaires à cette tâche sont importantes. De plus construire une ontologie représentant fidèlement le domaine concerné est toujours difficile. Comment en effet représenter des concepts qui peuvent être interprétés différemment entre les utilisateurs, ou quelle est la hiérarchie de concepts la plus objective ?

I.3.3. Identification thématique hiérarchique

L'adaptation des modèles de langage au thème du discours permettent une amélioration significative de la perplexité, ce sont des techniques issues du domaine de la recherche documentaire qui sont utilisées. Le problème soulevé est le fait que le thème est obtenu après une analyse de la totalité du document, ce qui n'est pas envisageable pour une adaptation thématique dynamique dans un système de RAP. Les travaux dans [15] se focalisent sur des méthodes statistiques qui auront pour motivation de déterminer au mieux le thème d'un document avec le minimum d'information possible. Dans ce cas précis les classificateurs Bayésiens de type unigrammes obtiennent les meilleures performances d'identification thématique, par rapport à des méthodes classiques. L'objectif est de résoudre les problèmes de différences de granularité entre les thèmes et de profiter des liens sémantiques entre les thèmes pour l'adaptation des modèles de langage.

L'identification thématique a pour but d'assigner un label thématique à un texte parmi un ensemble de labels possibles.

Dans [15] une approche d'identification thématique est présentée, cette approche exploite les relations sémantiques entre thèmes, par le biais d'une arborescence. L'utilisation des modèles de type unigrammes qui ont montré leur potentiel à discriminer les thèmes dans des applications thématiques est proposée. La proposition aussi des unigrammes thématiques hiérarchiques qui ont pour particularité, le fait que les relations entre frères sont favorisées dans le modèle, par l'attribution d'un vocabulaire commun. Par ailleurs, ce modèle utilisera son pouvoir discriminant

Partie I : Indexation audio

afin d'auto-évaluer sa décision thématique, en fournissant un degré de fiabilité sur le thème qu'il choisit.

I.4. Conclusion

Dans cette partie, nous avons présenté l'état actuel de recherches sur l'indexation audio. Ainsi, la segmentation audio peut consister en: la segmentation de la parole et de la musique, l'identification de langue, le suivi de locuteur, l'indexation de locuteur, l'indexation de mots-clés ou l'identification de thèmes.

Beaucoup de travaux sont nécessaires et doivent être effectués dans le domaine de l'indexation audio, en particulier, après l'extraction des descripteurs audio. Ces descripteurs doivent être organisés pour permettre la recherche et la navigation plus efficaces dans une base de données audio.

Dans la deuxième partie de ce chapitre nous allons nous intéresser à l'information musicale.

II. L'information musicale

II.1. La recherche de l'information musicale

La recherche d'information musicale relève d'une discipline appelée, dans le domaine de la recherche fondamentale dans les pays anglo-saxons, « Music Information Retrieval» ou « MIR ».

La recherche d'information musicale est un domaine de recherche multidisciplinaire qui s'efforce de développer des modèles innovants de recherche dans le contenu de bases de données ainsi que de nouvelles interfaces, et qui s'attache à faire évoluer les mécanismes de débit des réseaux afin d'en faire la plus vaste plate-forme commerciale musicale accessible à tous [16].

II.2. Typologie des documents musicaux

Dresser une typologie des documents musicaux revient à les classer selon les diverses représentations symboliques possibles du phénomène sonore. En effet, celui-ci, dans son abstraction (une série de vibrations aériennes, impalpables et immatérielles), peut prendre différentes formes.

On peut distinguer quatre sortes de représentation symbolique :

- une représentation acoustique: il s'agit de la forme par laquelle on peut visualiser le son en tant que manifestation physique. Ce type de représentation peut être l'image d'une courbe du signal sonore ou une formule mathématique (algorithme) par exemple;
- une représentation audio: autrement appelée enregistrement sonore, c'est la fixation sur un support des sons, ce support ayant pour finalité de permettre une écoute effective différée (en dehors de leur milieu originel) des sons;
- une représentation notée: partition, autrement dit le phénomène sonore organisé, sous sa forme graphique. La partition est principalement destinée à l'interprétation et permet également une écoute intérieure (comme on lit un livre) du phénomène sonore ;
- représentation textuelle: ce genre de représentation s'attache davantage à décrire le phénomène sonore qu'à le représenter physiquement.

II.3. Les facettes de l'information musicale

Une information musicale peut revêtir de multiples aspects: elle peut être constituée d'un ou plusieurs des paramètres du langage musical, ou bien être textuelle. D'autre part, les supports sur lesquels elle peut être fixée, en fonction de sa représentation symbolique, sont également multiples et interactifs. Enfin, une information musicale peut, dans une certaine mesure, être porteuse de sens. Afin d'en proposer une définition aussi complète que possible, J. Stephen Downie [17] attribue sept facettes à l'information musicale:

- ✓ tonale,
- ✓ temporelle,
- ✓ harmonique,
- ✓ timbrale,
- ✓ éditoriale,
- ✓ textuelle,
- ✓ bibliographique.

Elles peuvent être réparties en deux catégories :

- ✓ les facettes relevant du langage musical et regroupant les paramètres du son mais également les indications éditoriales ou textuelles exprimant le phénomène sonore ;
- ✓ les facettes connexes ou para-musicales, qui sont plutôt de l'ordre du complément (facette textuelle) ou des métadonnées (facettes bibliographique).

Ces facettes sont autant de caractéristiques qui, sans être exclusives les unes des autres, constituent des axes d'analyse non pas parallèles mais entrecroisés.

Une représentation graphique de ces différentes facettes est présentée ci-dessous

Partie II: L'information musicale

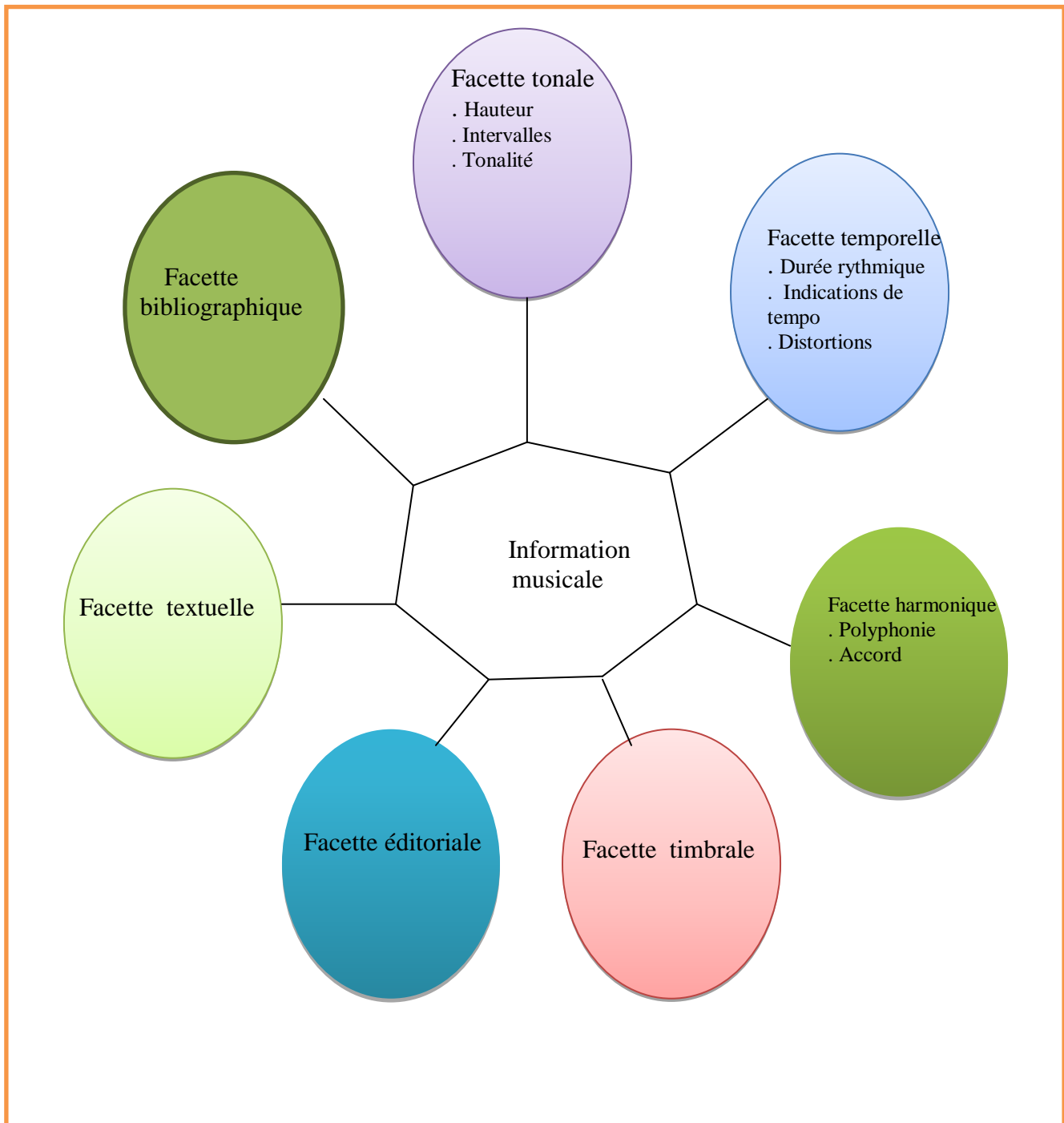


Figure 2 : Les sept facettes de l'information D'après [17]

L'information musicale est donc une notion riche et complexe, présentant de nombreux aspects constituant autant d'entrées pour une recherche d'information et, partant, de défis. Il convient de

Partie II: L'information musicale

les considérer dans leur ensemble afin de répondre au mieux aux besoins des utilisateurs et de pouvoir ainsi retrouver l'information utile.

II.4. Problèmes de l'indexation d'information Musicale

En comparaison avec les méthodes de recherche d'information textuelle, six problèmes majeurs se posent quand à la recherche d'information musicale [16, 17].

Parmi les problèmes de l'indexation musicale on trouve:

- Problèmes liés à l'indexation humaine de la musique
 - ✓ Représentation d'un contenu non textuel au moyen de textes
 - ✓ Absence de sujet (musique pure) ou sujet peu important pour la recherche
 - ✓ Plusieurs facettes à représenter: genre, émotion/atmosphère, forme musicale, instrumentation, thème/sujet, période, provenance, etc.
 - ✓ Plusieurs de ces facettes sont problématiques parce qu'elles sont hautement subjectives...
 - ✓ Caractère universel de la musique
- Les genres musicaux évoluent rapidement et de nouveaux genres apparaissent constamment qui sont définis à partir de critères musicaux, mais aussi sur plusieurs autres critères tels que :
 - ✓ Moyens de production
 - ✓ Personnalité de l'artiste
 - ✓ Fonction sociale et idéologique
- L'évolution des genres musicaux conduit aux résultats suivants :
 - ✓ Genres pas mutuellement exclusifs
 - ✓ Frontières entre genres flous et instables
 - ✓ Peu de recoupement entre les différentes taxonomies [18]
- Problèmes spécifiques à l'indexation automatique de la musique: La musique est créée par des humains, pour des humains, et les humains peuvent apporter une grande part de connaissances contextuelles dans tout ce qu'ils font; en fait, ils ne peuvent s'en empêcher et ils en sont rarement conscients. Les ordinateurs ne peuvent tenir compte de ces connaissances contextuelles ou très peu, et jamais sans avoir été spécifiquement programmés pour le faire. Conséquemment, faire quoi que ce soit avec la musique par ordinateur est très difficile; plusieurs problèmes sont essentiellement insolubles [19].
- Problèmes principalement liés à :

Partie II: L'information musicale

- ✓ Importance du contexte social et culturel dans l'appréciation et l'interprétation de la musique
- ✓ perception musicale
- ✓ mémoire musicale
- Plusieurs facettes à indexer: hauteur des sons, tempo, timbre, harmonie, texte (paroles), notes éditoriales [20].

II.4.1. Les représentations multiples de l'information musicale et sa complexité

La musique peut faire l'objet de multiples représentations symboliques, de même que les supports sur lesquels ils sont fixés. Dès lors, La complexité de l'information musicale peut également porter à conséquence sur les modalités de la recherche. La diversité des objectifs qui caractérise une requête d'ordre musical repose sur l'hypothèse généralement admise qu'une recherche d'information musicale est essentiellement concise, spécifique, unidimensionnelle et unidirectionnelle, comme l'est une recherche textuelle. De plus, les résultats potentiels de ce type de recherche sont supposés correspondre, tout au moins en partie, avec les termes contenus dans la requête [21].

Les paramètres du langage musical sont nombreux et imbriqués et ils peuvent être, mêlés à des éléments textuels, être répartis en facettes, formant ainsi une information musicale complexe. Ces paramètres constituent autant d'éléments distincts pouvant faire l'objet d'une recherche mais aussi pouvant servir de critères dans la formulation de requêtes élaborées. En cela, une recherche d'information musicale diffère d'une recherche d'information uniquement textuelle fondée, elle, principalement sur l'identification de chaînes de caractères alphanumériques.

Si les facettes textuelle et bibliographique peuvent être traitées comme objet d'une recherche textuelle, il en va autrement pour les éléments du langage musical. Une recherche portant sur ces derniers est donc spécifique certes, mais peut ne pas être concise. Elle n'est pas non plus unidimensionnelle, puisque tous les éléments du langage musical ainsi que les annotations manuscrites peuvent être des critères de recherche. Elle est encore moins unidirectionnelle, une recherche de ce type peut en effet porter aussi bien sur l'aspect horizontal du langage musical, autrement dit la mélodie, que sur son aspect vertical, à savoir l'harmonie. De plus, les éléments recherchés peuvent être spatialement dispersés au sein de plusieurs voix ou instruments dans le cadre d'une polyphonie.

Partie II: L'information musicale

II.4.2. La complexité des requêtes musicales

Les résultats d'une recherche musicale sont en partie déterminés par la façon dont la notion de donnée musicale a été préalablement définie. En effet, selon les personnes, une donnée musicale soit est nécessairement de provenance sonore et organisée dans le temps, soit est dépendante d'une notation et ordonnée spatialement [21]. Cette différence est cruciale dans une stratégie de recherche d'information musicale ainsi que dans les choix opérés lors de la conception d'une application de recherche spécifique à la musique. En effet, encoder telle représentation symbolique du phénomène sonore d'une seule façon implique que l'on élimine l'autre des stratégies de recherche. A l'inverse, prendre en compte les deux manières de considérer une donnée musicale afin d'obtenir des résultats les plus précis, efficaces et complets possibles est éminemment complexe et difficile pour ceux qui ont la charge de concevoir les outils nécessaires à la recherche d'information musicale. Notons enfin que l'usage du langage musical est fortement contextualisé et recèle une signification implicite [21]. En effet, le plus petit dénominateur commun en musique est la note, et elle est conditionnée par de multiples paramètres (hauteur, durée, etc.) il est nécessaire de prendre en compte chacune de ces notes au sein d'une pièce.

Le terme de donnée musicale est donc un peu plus vague. Les modèles de méthodes de recherche textuelles, unidimensionnelles et unidirectionnelles, ne peuvent être calquées tels quels dans le cadre d'une recherche d'information musicale puisqu'il faut considérer les nombreux paramètres et facettes de ce type de données.

Enfin, toujours en comparant recherche d'information textuelle et recherche d'information musicale, une autre question survient. En menant une recherche textuelle basée sur la sélection de mots-clés dans une base de données ou par internet, l'appariement entre requête formulée et résultats se fait sur la comparaison entre ces mots-clés et les termes reconnus dans un contenu. Et cela, le plus souvent, au détriment d'autres termes non porteur d'une signification. Ce sont les mots vides, qui la plupart du temps font l'objet d'une liste contrôlée. Or, dans le périmètre d'une recherche d'information musicale non textuelle, on peut s'interroger sur l'existence ou non de mots vides dans le langage musical et, s'ils existent, sur les formes qu'ils peuvent prendre.

Répondre aux besoins d'information musicale des utilisateurs revient par conséquent à se poser une question qui peut sembler triviale: quel est le véritable objet sur lequel porte la recherche d'information musicale ?

Des outils de recherche prenant en compte la totalité des paramètres et facettes musicaux dans le choix des critères de formulation des requêtes sont difficile à réaliser.

Partie II: L'information musicale

II.4.3. L'évaluation des résultats d'une recherche d'information musicale

Dans le cadre d'une recherche d'information textuelle, l'appariement entre la requête et les résultats de cette requête se mesure à l'aide d'indicateurs normalisés.

Les deux principaux sont :

- ❖ le taux de précision, soit le rapport entre le nombre de documents pertinents trouvés et le nombre total de documents trouvés;
- ❖ le taux de rappel, ou rapport entre le nombre de documents pertinents trouvés et le nombre total de documents parmi lesquels a été effectuée la recherche.

Ces indicateurs ont pour fonction d'évaluer la pertinence des documents et informations trouvés. Cependant, cette notion de pertinence reste, dans l'abstraction, imprécise et ce malgré la normalisation des procédés de sa mesure. En effet, on ne peut évaluer la pertinence des résultats qu'à propos d'une recherche portant sur un objet particulier menée par l'intermédiaire d'une requête formulée d'une certaine façon et dans un environnement donné (base de données, collection de documents, internet, etc). De manière générale, on considère comme pertinent un document ou une information qui est du même sujet énoncé dans la requête [16].

Dans le cadre d'une recherche d'information musicale, il est à craindre que cette notion de pertinence soit encore plus floue, voire inadaptée [16], ceci pour deux raisons. La première est qu'une recherche d'information musicale, peut porter sur quantité de supports, de représentations symboliques et de paramètres différents. La seconde raison tient à la subjectivité dont chacun fait preuve face à la musique.

On peut le constater lors d'une recherche par connotation ou « mood », ou même encore au sein d'un réseau social comme un forum de discussion ou une page web personnelle, dans lesquels il peut y avoir une indexation par « tags ». Musique et sons sont alors représentés par un langage documentaire reposant sur des associations avec des informations sensorielles: images, couleurs, luminosités, mouvements, sentiments, etc. qui peuvent être partagées ou non de façon communautaire ou interculturelle.

L'appariement entre la requête et les résultats devient alors problématique puisque la pertinence de ces derniers est évaluée selon le goût de l'utilisateur et la résonance psychologique qu'ils peuvent susciter chez lui.

Partie II: L'information musicale

II.5. Méthode et outils de recherche d'information Musicale

II.5.1. Méthodes et offre d'outils disponibles

Un certain nombre d'outils sont disponibles pour la recherche d'information musicale. La plupart sont issus du monde de la recherche scientifique (centres de recherches universitaires, pôles de recherche et développement d'entreprises, etc.) et ne sont, pour le moment, que peu exploités à grande échelle de façon commerciale ou même institutionnelle.

Il suffit, pour s'en rendre compte, de naviguer sur les sites web de fournisseurs de contenu destinés au grand public (iTunes Music Store, Fnac, Di-arezzo, etc.) ou de centre de ressources documentaires (Cité de la musique). On constatera alors que les procédés mis à disposition pour retrouver des titres audio ou des partitions par exemple demeurent traditionnels. Ils restent basés sur le principe d'une recherche bibliographique ne proposant comme critères de recherche que des champs du type: titre œuvre ou titre chanson, artiste interprète, etc. ou par classification: recherche par genre musical par exemple. Les fournisseurs de contenu ayant mis en place des outils de recherche spécifiques dépassant le stade de la recherche par métadonnées restent marginaux.

II.5.2. Exemples de méthodes et d'outils

- La recherche de partitions par internet en utilisant un moteur de recherche généraliste.
- La recherche par chantonnement ou query by humming: l'utilisateur formule sa requête en chantant ou sifflant tout ou partie d'une mélodie dont il souhaite retrouver le titre ou l'œuvre de laquelle elle est tirée. Ce type de recherche, repose sur un principe de reconnaissance vocale et s'effectue essentiellement dans des bases de données. Il constitue une alternative à la recherche textuelle ou bibliographique, par titre, compositeur ou interprète par exemple. Les outils ou systèmes de recherche de musique par chantonnement actuellement disponibles sont: naiyo¹, Midomi².
- La recherche d'exemples de musique: l'utilisateur présente un exemple du résultat de recherche attendu. Le système recherchera alors toutes les musiques qui correspondent à cet exemple. Les outils ou systèmes de recherche d'exemple de musique actuellement disponibles sont: ThemeFinder³ et Musipedia⁴.

¹ www.naiyo.com

² www.midomi.com

³ www.themefinder.org

⁴ www.musipedia.org

Partie II: L'information musicale

- La recherche par tags: c'est une recherche basée sur le principe de la similarité. Le site de la société Shazam⁵ propose à ses utilisateurs, par l'intermédiaire de profils personnels, de «tagger» leurs artistes favoris. Un module de recherche par artiste est proposé et les résultats des requêtes s'affichent par le biais de «tags» sous la forme d'une représentation graphique.

II.6. Revue de la littérature

La recherche d'information musicale est un domaine complexe qui contient de multiples facettes. Elle est très dépendante de la perception selon la zone géographique et culturelle à laquelle nous appartenons. De nombreuses méthodes de recherche d'informations musicales ont été proposées. L'objectif est de créer des systèmes qui indexent et recherchent de façon efficace des chansons dans de grandes bases de contenu musical.

La façon la plus courante d'indexer et de recherche de la musique utilise des métadonnées qui sont la seule facette d'information musicale qui ne provient pas du contenu d'une composition. La recherche de la musique est faite à travers une combinaison de métadonnées factuelles et/ou culturelles tels que Gracenote⁶, MusicBrainz⁷ et Pandora⁸. Les métadonnées factuelles sont des métadonnées objectives qui contiennent des informations à propos de l'œuvre musicale (telles que le titre, l'artiste, l'album, l'année de publication et la durée). Les métadonnées culturelles contiennent des concepts subjectifs (tels que le genre, le style, l'humeur et l'émotion). Les métadonnées sont bien expressives et peuvent satisfaire les utilisateurs. Cependant, Aucouturier dans [18] et Downie dans [17] affirment que les métadonnées peuvent être utilisées comme entrées des systèmes afin d'aider les utilisateurs pour effectuer la recherche de musiques. Néanmoins, l'inconvénient de l'utilisation des métadonnées est qu'une nouvelle chanson doit être annotée manuellement avant d'être recherchée. En outre, il est extrêmement difficile de maintenir des descriptions de métadonnées expressives cohérentes quand les catalogues contiennent plus de milliers de morceaux. Howard dans [22] et Temperley dans [23] exposent les difficultés liées aux métadonnées bibliographiques.

Une autre façon pour la recherche d'information musicale est d'effectuer une recherche sans avoir aucune information seulement une idée. Dans ce cas, l'utilisateur peut chanter, fredonner ou

⁵ www.shazam.com

⁶ www.gracenote.com

⁷ www.musicbrainz.org

⁸ www.pandora.com

Partie II: L'information musicale

donner un exemple de la chanson. La recherche d'information musicale offre la possibilité de récupérer des chansons en décrivant et analysant leurs contenus. Les systèmes de recherche d'œuvres musicales basés-contenu ont été présentés par Goto dans [25]. Comme par exemple Query by Example (QBE) qui est un type d'interface utilisateur servant à effectuer des recherches dans des bases de données relationnelles. Le principe d'une interface QBE est que l'utilisateur présente une requête sous forme d'un exemple du résultat de recherche attendu, puis le soumet au SGBD. Le système mesure la similarité entre la requête et toutes les chansons dans la base de données. Une limitation de cette approche est qu'elle oblige à avoir un exemple audio afin de spécifier une requête. Pour les cas dans lesquels un tel exemple n'est pas disponible, Dannenberg [24] et Phiwma [26] ont développé des requêtes par fredonnement query by Hamming. L'utilisation de cette approche est difficile, surtout pour les utilisateurs non formés qui n'ont pas d'expérience pour émuler le tempo, la hauteur, la mélodie, et le timbre [24].

Une autre voie de recherche d'information musicale s'est concentrée sur les propriétés acoustiques de la musique et qui utilise des approches permettant l'extraction des caractéristiques musicales en utilisant les techniques de traitement du signal [27, 28, 29, 30, 31, 32, 33]. Cependant, la complexité des signaux musicaux peut manquer des informations utiles ou peut fournir des informations erronées.

Il est aussi possible que l'utilisateur n'ait qu'un extrait de parole. Dans ce cas, il existe des systèmes de recherche de musique comme musikiwi⁹ et paroles¹⁰ qui offrent la possibilité de récupérer des chansons en remplissant quelques mots de paroles et intègrent un moteur de recherche afin de permettre la recherche par mots-clés en entrant une requête [34, 35, 36]. Ces systèmes de recherche d'information musicale ne peuvent pas résoudre complètement les problèmes de recherche de musique, cela est dû aux techniques d'indexation pauvres qu'ils utilisent.

Une chanson est une œuvre musicale composée d'un texte (parole) et d'une mélodie destinée à être interprétée par la voix humaine. Dans la suite de la thèse, on combine les métadonnées factuelles, culturelles et les paroles pour indexer et rechercher une chanson, puisque à notre avis, n'importe quel système musical est incomplet s'il ne tient pas compte de la dimension du contenu sémantique des chansons. Une grande partie de la sémantique d'une œuvre musicale réside

⁹ www.musikiwi.com

¹⁰ www.paroles.fr

Partie II: L'information musicale

exclusivement dans ses paroles [37]. Plusieurs travaux ont été réalisés en utilisant les paroles pour la recherche des chansons, l'analyse des paroles de chansons complète ce qui peut être tirés de l'analyse culturelle, collaborative ou acoustique [38]. Une grande partie de recherches musicales se réfère directement aux paroles [39]. Le contenu de paroles de chansons peut déterminer le thème d'une chanson [40, 41], son genre [34, 42] et peut être utilisé pour l'indexation de la musique et pour les l'artistes similaires [45]. Actuellement, les paroles sont devenues des sources de génération des métadonnées [20], sans doute inspiré par l'évolution du Web 2.0, les paroles ont été trouvées comme une base utile de génération de mots-clés pour les chansons, une technique qui peut finalement conduire à un étiquetage automatique [46].

Knees et al présentent les étapes de pionner pour la construction d'un système automatique de recherche, qui est capable de trouver la musique qui satisfait les requêtes arbitraires en langage naturel [43]. Un autre travail utilise une classification d'humeur des chansons est décrite dans [44]. Ces modèles sont basés sur des techniques traditionnelles qui reposent sur le traitement syntaxique de mots-clés et ignorent le côté sémantique qui rend les résultats plus pertinents.

La représentation approximative, pauvre et partielle du contenu sémantique des chansons en utilisant des techniques d'indexation traditionnelles à conduit à l'utilisation des formalismes de représentation de connaissances plus précises et plus riches. Parmi ces formalismes, les ontologies qui permettent de caractériser un domaine par un ensemble de concepts et de relations entre eux. Dans la littérature, quelques études ont utilisés les ontologies pour la représentation de la musique [47, 48, 49, 50, 51].

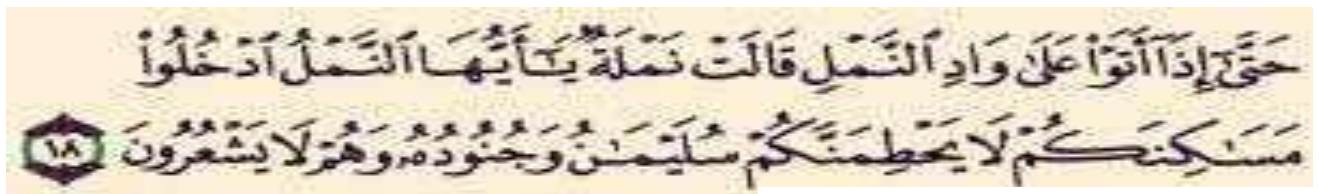
II.7. Conclusion

L'information musicale est une notion riche et complexe, présentant de nombreux aspects ou facettes (tonale, temporelle, harmonique, timbrale, éditoriale, textuelle, bibliographique) constituant autant d'entrées pour une recherche d'information. Décrire et représenter par des métadonnées (facette bibliographiques) une œuvre musicale (chanson) notamment sous sa forme textuelle (parole: facette textuelle). C'est ce qui occupe le deuxième chapitre de cette thèse qui consiste à indexer une collection de chansons, en les représentant sous forme de vecteurs de descripteurs afin de les comparer, de mesurer leurs distances et de répondre à des requêtes de différents natures. La catégorisation supervisée qui est un axe de l'indexation est réalisée en utilisant premièrement l'algorithme Bayes naïf puis en adaptions un algorithme de colonie de fourmis.

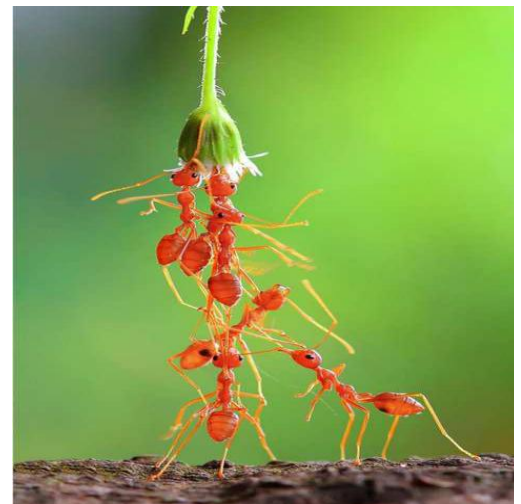
Chapitre II: Swarm Intelligence et colonies de fourmis pour la catégorisation de chansons

Chapitre II

Le Quotidien britannique « The Times » a publié un article le 6 février 2009, sous le titre (Hills are alive with the sound of ants - talking to each other) pour les fourmilières vivantes avec des voix, en parlant avec les uns des autres). L'article parle des nouvelles découvertes scientifiques sur le langage de communication et la conversation dans le royaume de fourmis. L'article mentionne que les découvertes récentes ont montré que la langue de communication chez les fourmis est sophistiquée et avancée significativement plus qu'on ne le pensait auparavant. En insérant des microphones et des enceintes miniatures dans les nids, les chercheurs ont découvert que la reine pouvait donner des instructions à ses ouvrières. Ils ont enregistré ses "discours", et ont également découvert que d'autres insectes imitent ces messages, pour faire des fourmis leurs esclaves. Ainsi, les chenilles d'une espèce de papillon détournent les fourmis de leur tâche pour se faire nourrir. Il a été indiqué dans le Saint Coran :



Quand ils arrivèrent à la Vallée des Fourmis, une fourmi dit: "O fourmis, entrez dans vos demeures, (de peur) que Salomon et ses armées ne vous cassent (écrasent) sans s'en rendre compte.



*Si nous prenons la nature pour guide,
nous ne nous égarerons jamais.*

[Cicéron]

Chapitre II

Ce chapitre est divisé en trois parties: la première partie traite la catégorisation de chansons. La seconde partie est consacrée à l'intelligence en essaim et détaille les colonies de fourmis tandis que la troisième partie expose la catégorisation de chansons en utilisant l'algorithme de colonie de fourmis.

Dans la première partie de ce chapitre, nous exposons la représentation de chansons, nous définissons la notion de catégorisation de chansons en utilisant l'algorithme Bayes naïf et en fin, nous présentons les critères de mesure de performances.

L'objectif de la seconde partie de ce chapitre est de présenter les concepts de bases de la swarm Intelligence ainsi que les modèles et les techniques qui lui sont associées. Nous commençons par la définition du comportement biologique collectif et son principe, puis, nous abordons la swarm Intelligence à travers ses modèles, ses principes et ses techniques, ensuite nous présentons quelques uns de ses avantages et domaines d'application puis nous présentons les différents concepts liés à l'optimisation par colonies de fourmis ainsi que l'algorithme ACO associé à cette approche et son utilisation en Data Mining et, plus précisément, pour la classification de textes.

La troisième partie de ce chapitre présente notre contribution au problème posé dans cette thèse, à savoir la proposition d'une approche de catégorisation de chanson qui consiste à adapter la métaheuristique ACO pour la classification supervisée de chansons.

Partie I : Catégorisation de chansons

I. Catégorisation de chansons

I.1. Introduction

Dans cette partie, nous exposons la catégorisation de chansons en s'inspirant des méthodes de classifications textuelles. Nous donnons une définition de la catégorisation puis nous présentons la représentation et le codage d'une chanson. Nous utilisons l'algorithme Bayes naïf pour la classification de chansons par thème basées sur les métadonnées en utilisant le titre, le refrain et les caractéristiques de l'artiste. Nous proposons d'utiliser le titre de la chanson qui décrit brièvement son thème et le refrain qui est la partie qui se répète au moins deux fois et que les auditeurs ont tendance à se rappeler. Le refrain clarifie la signification de la chanson. On utilise aussi les caractéristiques de l'artiste, puisque chaque artiste a tendance à composer ou à chanter un genre particulier de musique. En fin, nous présentons les critères de mesure de performances des classificateurs.

I.2. Catégorisation de chansons selon le texte

La recherche dans le domaine de catégorisation automatique est toujours d'actualité, car les résultats obtenus aujourd'hui sont encore sujets d'améliorations. Pour certaines tâches, les classificateurs automatiques performant presque aussi bien que les humains, mais pour d'autres, l'écart est encore grand. Au premier abord, l'essentiel du problème est facile à saisir. D'un côté, on est en présence d'une banque de chansons et de l'autre, d'un ensemble prédéfini de catégories. L'objectif est de rendre une application informatique capable de déterminer de façon autonome, dans quelle catégorie classer chacune des chansons.

Pour résoudre le problème de catégorisation de chansons, nous utilisons les informations verbales. Nous allons nous concentrer sur les mots constituant le titre et les refrains de la chanson pour déterminer le thème de cette dernière et les caractéristiques de l'artiste pour déterminer le genre de la musique. Nous présentons le classificateur Bayes naïf et nous l'adaptions pour la catégorisation de chansons comme classificateur de référence.

I.2.1. Définition de la catégorisation

Le but de la catégorisation automatique de chansons est d'apprendre à une machine à classer une chanson dans la bonne catégorie en se basant sur son contenu, les catégories font référence aux thèmes (sujets). Il se peut qu'on désire qu'une même chanson ne soit associée qu'à une seule

Partie I : Catégorisation de chansons

catégorie ou bien on peut se permettre que plusieurs catégories accueillent une même chanson. L'ensemble de catégories possibles est déterminé à l'avance. Le problème consiste à regrouper les chansons selon leurs similarités. Dans une catégorisation de chansons: la classification s'apparente au problème de l'extraction de la sémantique des mots constituant le titre et le refrain de la chanson, puisque l'appartenance de la chanson à une catégorie est étroitement liée à la signification des mots constituant le titre et le refrain de la chanson.

I.2.2. Comment catégoriser une chanson?

Le processus de catégorisation intègre la construction d'un modèle de prédiction qui, en entrée reçoit le titre, le refrain de la chanson et les caractéristiques de chansons et, en sortie, lui associe une ou plusieurs étiquettes.

Pour identifier la catégorie ou la classe à laquelle une chanson est associée, un ensemble d'étapes est habituellement suivies. Ces étapes concernent principalement:

- La manière dont une chanson est représentée
- Le choix de l'algorithme d'apprentissage à utiliser.
- Comment évaluer les résultats obtenus pour garantir une bonne généralisation du modèle appris.

Le processus de catégorisation, intégrant la phase de classement de nouvelles chansons comporte deux phases que l'on peut distinguer comme suit:

- 1- L'apprentissage, qui comprend plusieurs étapes et aboutit à un modèle de prédiction.
- 2- Le classement d'une nouvelle chanson.

I.2.3. Représentation et codage d'une chanson

Les algorithmes d'apprentissage ne sont pas capables de traiter directement les textes, plus généralement, les données non structurées comme les images, les sons et les séquences vidéo. C'est pourquoi une étape préliminaire dite représentation est nécessaire. Cette étape consiste généralement, dans notre travail, en la représentation de chaque chanson par un vecteur dont les composantes sont les mots contenus dans le titre et le refrain de la chanson. Une collection de chansons peut être ainsi représentée par une matrice A , dont chaque élément représente les occurrences du terme qui apparaît dans le titre et le refrain de la chanson

$$A = (w_{ij}) \quad (1)$$

Où w_{ij} est le poids du terme j dans la chanson i .

Partie I : Catégorisation de chansons

On s'inspire du travail dans [51] et on transforme l'ensemble de chansons en un tableau croisé (matrice) *individus-variables*.

L'individu est une chanson S_j , étiquetée lors de la phase d'apprentissage, elle est classée dans la phase de prédiction. Les variables sont les descripteurs (les termes) t_k . Le contenu du tableau (les éléments w_{kj}), au croisement de la chanson j et du terme k , représente le poids de ce terme k dans la chanson j .

On transforme la chanson S_j (titre et refrain de chansons) en un vecteur $S_j = (w_{1j}, w_{2j}, \dots, w_{|T|j})$, où T est l'ensemble de termes (descripteurs) qui apparaissent au moins une fois dans le corpus (la collection) d'apprentissage.

Les poids w_{kj} correspondent à la contribution des termes t_k à la sémantique de la chanson S_j .

Différentes méthodes sont proposées pour le choix des descripteurs et des poids associés à ces descripteurs. Certains chercheurs utilisent, à titre d'exemples, les mots comme descripteurs, tandis que d'autres préfèrent utiliser les lemmes (racines lexicales) [52, 53] ou encore des *stemmes* (la suppression d'affixes).

On utilise le codage TF X IDF pour coder nos chansons. Le codage TF X IDF a été introduit dans le cadre du modèle vectoriel, il donne beaucoup d'importance aux mots qui apparaissent souvent à l'intérieur d'un même texte, ce qui correspond bien à l'idée intuitive que ces mots sont plus représentatifs du document. Mais sa particularité est qu'il donne également moins de poids aux mots qui appartiennent à plusieurs documents: pour refléter le fait que ces mots ont un faible pouvoir de discrimination entre les classes [54].

Le poids du terme w_{ij} dans une chanson S_j est calculé ainsi :

$$w_{ij} = tf_{ij} * \log\left(\frac{N}{sf_j}\right) \quad (2)$$

Où tf_{ij} est la fréquence du terme j dans la chanson i , ou le nombre d'occurrences du terme j dans la chanson i . N est le nombre total de chansons et sf_j est la fréquence de chansons, qui est le nombre de chansons dans lesquelles, le terme j apparaît au moins une fois. La fréquence inverse de chansons (IDF) est donné par: $\log\left(\frac{N}{sf_j}\right)$.

Une fois normalisé, nous ajoutons aux vecteurs les caractéristiques des chanteurs (le nom de l'artiste, le thème de chanson et le genre de musique qu'il chante en général).

Partie I : Catégorisation de chansons

Une fois que les chansons sont représentées sous forme de vecteurs, la similarité entre deux chansons peut être obtenue avec la distance cosinus défini comme suit:

$$\text{cosine}(S_i, S_j) = \frac{\sum_{k=1}^n (w_{ik} * w_{kj})}{\sqrt{\sum_{k=1}^n (w_{ik})^2 * \sum_{k=1}^n (w_{kj})^2}} \quad (3)$$

I.3. Catégorisation de chansons en utilisant l'algorithme Bayes naïf

En apprentissage automatique, différents types de classificateurs ont été mis au point pour atteindre un degré maximale de précision et d'efficacité, chacun ayant ses avantages et ses inconvénients. Mais, ils partagent toutefois des caractéristiques communes.

Il existe plusieurs algorithmes d'apprentissage automatique: l'algorithme de Rocchio, les réseaux de neurones, les machines à support vectoriel, l'algorithme des k-voisins les plus proches et le classificateur Bayésien naïf; algorithme surpassé par d'autres mais il est souvent utilisé comme point de référence à cause de sa simplicité.

Comme son nom l'indique, le classificateur Bayésien naïf se base sur le théorème de Bayes permettant de calculer les probabilités conditionnelles. Dans un contexte général, ce théorème fournit une façon de calculer la probabilité conditionnelle d'une cause sachant la présence d'un effet, à partir de la probabilité conditionnelle de l'effet sachant la présence de la cause ainsi que des probabilités a priori de la cause et de l'effet. Dans notre travail, on applique l'algorithme Bayes naïf à la catégorisation de chansons en utilisant le titre, le refrain de la chanson, et les caractéristiques de l'artiste pour déterminer la catégorie de la chanson.

On peut résumer l'utilisation du classificateur Bayésien naïf lorsqu'il est appliqué à la classification de chansons ainsi:

Les caractéristiques de l'artiste sont importantes pour déterminer la catégorie de la chanson car chaque artiste a tendance à composer ou chanter un genre de musique. En plus pour le titre et le refrain on procède de la façon suivante

- On cherche la classification qui maximise la probabilité d'observer les mots du titre et du refrain d'une chanson.
- Lors de la phase d'apprentissage, le classificateur calcule les probabilités qu'une nouvelle chanson appartienne à telle catégorie à partir de la proportion des chansons d'apprentissage appartenant à cette catégorie. Il calcule aussi la probabilité qu'un mot donné soit présent

Partie I : Catégorisation de chansons

dans le titre ou le refrain d'une chanson, sachant que cette chanson appartient à telle catégorie.

- Par la suite, quand une nouvelle chanson doit être classée, on calcule les probabilités qu'elle appartienne à chacune des catégories à l'aide de la règle de Bayes et des valeurs calculées à l'étape précédente.

$$\begin{aligned}c_{select}(C) &= \arg_{c \in C} \max P(c \setminus L, art) \\ &= \arg_{c \in C} \max P(c \setminus L) * P(c \setminus art)\end{aligned}\tag{4}$$

$P(c \setminus art)$ est estimé par la formule suivante:

$$P(c \setminus art) = \left(\frac{N(art, c)}{N(art)} \right)\tag{5}$$

$N(art)$ est le nombre de chansons de l'artiste art et $N(art, c)$ est le nombre de chansons de l'artiste art avec le thème c .

$P(c \setminus L)$ est estimé par la formule suivante ou L représente les mots du titre et du refrain

$$P(c \setminus L) = p(c) * p(L \setminus c)\tag{6}$$

La probabilité à estimer est donc $P(c_j \setminus a_1, a_2, a_3, \dots, a_n)$

Où c_j est une catégorie et a_i est un attribut.

A l'aide du théorème de Bayes, on obtient

$$P(c_j \setminus a_1, a_2, a_3, \dots, a_n) = \frac{P(a_1, a_2, a_3, \dots, a_n \setminus c_j) * P(c_j)}{P(a_1, a_2, a_3, \dots, a_n)}\tag{7}$$

On peut omettre de calculer le dénominateur, qui reste le même pour chaque catégorie.

En guise de simplification, on calcule $P(a_1, a_2, a_3, \dots, a_n \setminus c_j)$ ainsi

$$P(a_1, a_2, a_3, \dots, a_n \setminus c_j) = \prod_{i=1}^n P(a_i \setminus c_j)\tag{8}$$

La probabilité qu'un mot apparaisse dans une chanson est indépendante de la présence des autres mots de la chanson. On sait que cela est faux. Pourtant, cette supposition n'empêche pas un tel classificateur de présenter des résultats satisfaisants. Et surtout, elle réduit de beaucoup les calculs nécessaires. Sans elle, il faudrait tenir compte de toutes les combinaisons possibles de mots, ce qui d'une part impliquerait un nombre important de calculs, mais aussi réduirait la qualité statistique

Partie I : Catégorisation de chansons

de l'estimation, puisque la fréquence d'apparition de chacune des combinaisons serait très inférieure à la fréquence d'apparition des mots pris séparément [51].

Pour estimer la probabilité $P(a_i|c_j)$, on pourrait calculer directement dans les chansons d'apprentissage la proportion de ceux appartenant à la classe c_j qui contiennent le mot a_i . Cependant, l'estimation ne serait pas très valide pour des numérateurs petits.

Dans le cas extrême où un mot ne serait pas du tout rencontré dans une classe, sa probabilité de 0 dominerait les autres dans le produit ci-dessus et rendrait nulle la probabilité globale. Pour pallier ce problème, une bonne façon de faire est d'utiliser le m-estimé qui est calculé ainsi :

$$\frac{n_k+1}{n+|\text{vocabulaire}|}$$

Où n_k est le nombre d'occurrence du terme dans la classe c_j , n est le compte total des mots dans le corpus d'apprentissage et $|\text{vocabulaire}|$ représente le nombre de mots-clés.

I.4. Critères de mesure de performances des classificateurs

Lorsqu'une personne interroge une base de données (que ce soit un logiciel documentaire ou un moteur de recherche), elle attend un nombre de réponses (sous forme de documents) supérieur ou égal à un. À partir de l'ensemble de réponses obtenues mis en regard de l'attente de l'utilisateur, on peut mesurer les performances de l'algorithme de recherche mis en œuvre pour retrouver un document. Les critères de mesure des performances sont le rappel, la précision et la F-mesure.

I.4.1. Le rappel

Le rappel est défini par le nombre de chansons pertinentes retrouvées au regard du nombre de chansons pertinentes que possède la base de données. Cela signifie que lorsque l'utilisateur interroge la base il souhaite voir apparaître toutes les chansons qui pourraient répondre à son besoin. Si cette adéquation entre le questionnement de l'utilisateur et le nombre de chansons présentées est importante alors le taux de rappel est élevé. À l'inverse si le système possède de nombreux chansons intéressantes mais que celles-ci n'apparaissent pas dans la liste des réponses, on parle de silence. Le silence s'oppose au rappel.

$$\text{rappel}_i = \frac{\text{nb de chansons correctement attribuées à la classe } i}{\text{nb de chansons appartenant à la classe } i} \quad (9)$$

Partie I : Catégorisation de chansons

I.4.2. La précision

La précision est le nombre de chansons pertinentes retrouvées rapporté au nombre de chansons total proposé par le moteur de recherche pour une requête donnée.

Le principe est le suivant: quand un utilisateur interroge une base de données, il souhaite que les chansons proposées en réponse à son interrogation correspondent à son attente. Toutes les chansons retournées superflues ou non pertinentes constituent du bruit. La précision s'oppose à ce bruit. Si elle est élevée, cela signifie que peu de chansons inutiles sont proposées par le système et que ce dernier peut être considéré comme "précis". On calcule la précision avec la formule suivante :

$$\text{précision}_i = \frac{\text{nb de chansons correctement attribuées à la classe } i}{\text{nb de chansons attribuées à la classe } i} \quad (10)$$

I.4.3. F-mesure

Une mesure populaire qui combine la précision et le rappel, nommée F-mesure (soit *F-measure* en anglais) ou F-score. Elle est également connue sous le nom de mesure F_1 , car précision et rappel sont pondérés de façon égale :

$$F = \frac{2 * (\text{précision} * \text{rappel})}{(\text{précision} + \text{rappel})} \quad (11)$$

I.5. Conclusion

Dans cette partie, nous avons vu que dans une catégorisation de chansons, la classification s'apparente au problème de l'extraction de la sémantique des mots constituant le titre et le refrain de chansons, puisque l'appartenance de la chanson à une catégorie est étroitement liée à la signification du titre, refrain et les caractéristiques de l'artiste de la chanson. Nous avons catégorisé une banque de chansons en utilisant l'algorithme Bayes naïf. Nous avons exposé les critères de performances qui seront utilisés pour évaluer les approches Bayes naïf et l'algorithme de colonie de fourmis pour la catégorisation de chansons objet de la partie trois.

Partie II Swarm Intelligence et colonies de fourmis

II. Swarm Intelligence et colonies de fourmis

II.1. Introduction

Dans le domaine de l'informatique, depuis ses débuts, il a toujours semblé intéressant de s'inspirer de phénomènes, de comportements sociaux ou collectifs chez certaines espèces vivantes. Malgré la richesse des idées dans plusieurs disciplines telles que la zoologie, les moyens de calculs de l'époque antérieure aux années 90 étaient vraiment insuffisants et les problèmes qui surgissaient devenaient vite insurmontable.

De nos jours, précisément durant cette dernière décennie, et avec l'apparition de plusieurs défis (masses gigantesques de données, système distribués, modélisation et extraction de connaissance), plusieurs chercheurs en informatique ont orienté leurs recherches vers des approches très intéressantes qui s'inspirent directement de la nature. C'est le cas, par exemple, des insectes sociaux, ces derniers arrivent à survivre depuis des millions d'années dans des conditions très difficiles et un environnement imprévisible. Ces insectes possèdent des caractéristiques remarquables telles que l'autonomie, l'auto-organisation, l'optimisation, le contrôle décentralisé..., ce qui a attiré beaucoup de chercheurs vers la voie bio-inspirée afin de résoudre des problèmes complexes tels que ceux de l'optimisation. Diverses techniques bio-inspirées ont ainsi vu le jour, telles que les réseaux de neurones artificiels inspirés de l'architecture du cerveau humain ainsi que les algorithmes génétique qui ont fait leurs preuves dans la résolution de divers types de problèmes.

La swarm intelligence ou intelligence en essaim est une discipline bio-inspirée qui s'intéresse particulièrement à l'intelligence qui émerge à partir de comportement collectifs chez certaines espèces, vivant dans des groupes et possèdent des propriétés intéressantes, et essaye d'adapter ces propriétés aux problèmes réels.

II.2. Intelligence collective dans la nature

Dans la nature et presque chez tous les insectes ou les animaux qui vivent ensemble dans des groupes, on remarque une sorte d'intelligence qui émerge à partir du comportement collectif, et social. Le niveau d'intelligence varie selon l'espèce (fourmis, abeilles, oiseaux, poissons...) et l'environnement et le niveau d'interaction et d'autres facteurs, nous essayons dans ce qui suit d'éclairer un peu les idées concernant la swarm intelligence par l'illustration de quelques exemples.

Partie II Swarm Intelligence et colonies de fourmis

C'est un fait bien connu qu'une fourmi individuelle n'est pas lumineuse et presque aveugle mais les fourmis dans une colonie travaillant en équipe, font des choses remarquables comme l'exploration (foraging) efficace, tri optimal de couvée ou formation impressionnante de cimetière des fourmis mortes. D'autres espèces biologiques, tels que les poissons, les termites, les guêpes, les moutons, les oiseaux, les chats et les criquets présentent un comportement collectif intelligent bien qu'elles se composent d'individus simples. En informatique, on appelle ces individus agents.

II.3. Particularités du comportement collectif

L'organisation des populations en essaim est caractérisée par un phénomène de groupes, dont les particularités sont les suivantes:

- ✓ *Information locale limitée:* Chaque individu possède une connaissance locale de l'environnement et ne connaît rien de la fonction globale de la population.
- ✓ *Des règles individuelles simples:* Chaque individu possède un jeu de règles comportementales simples limitées. Ce jeu de règles permet au groupe de coordonner ses actions collectivement et de construire une structure ou d'adopter une configuration globale.
- ✓ *La structure globale émergente accomplit une certaine fonction:* Ces structures permettent au groupe de résoudre un problème. Pour cela, ces structures sont flexibles (adaptatives à l'environnement) et robustes (adaptatives aux erreurs de certains individus).

II.4. Introduction à l'intelligence en essaim (swarm intelligence SI)

L'intelligence en essaim (ou Swarm Intelligence: SI) est une discipline d'intelligence artificielle moderne qui est concernée, entre autres, par la conception de systèmes multi-agents. Le paradigme de conception pour ces systèmes est fondamentalement différent d'approches plus traditionnelles [55].

Ainsi, l'intelligence en essaim est une voie informatique innovatrice pour la résolution des problèmes difficiles. En général, cette résolution est inspirée du comportement des créatures biologiques dans leurs essaims et colonies [57].

Elle est considérée comme une métaphore comportementale pour résoudre des problèmes distribués inspirés d'exemples biologiques fournis par des insectes sociaux comme les fourmis, les termites, les abeilles ou les guêpes et par essaim, troupeau, masse et phénomènes de bancs chez les animaux comme les bancs de poissons et les masses d'oiseaux [58].

Partie II Swarm Intelligence et colonies de fourmis

L'observation de populations animales, et plus particulièrement celles d'insectes a amené les chercheurs à comprendre les mécanismes sociaux qui permettent à ces individus de coordonner leurs actions, afin d'obtenir la meilleure efficacité possible. Les mécanismes mis en jeu ont permis d'élaborer, dans le domaine de la robotique, des stratégies coopératives entre différents robots autonomes [58].

La Swarm Intelligence est basée sur les principes qui sont à la base du comportement de systèmes naturels composé de beaucoup d'agents, l'exploitation des formes de communication est locale et le contrôle est fortement distribué [58].

Au lieu de concevoir des systèmes complexes et centralisés, de nos jours, les concepteurs préfèrent plutôt travailler avec plusieurs agents simples et autonomes. Beaucoup d'agents coopérant dans le système peuvent résoudre des problèmes très complexes avec un effort de conception minimal. En général, les systèmes multi-agents qui utilisent l'intelligence en essaim sont appelés des systèmes intelligents à base d'essaim (swarm intelligent systems). Ils sont surtout utilisés comme moteurs de recherche et outils d'optimisation [57].

Par exemple, les fourmis, les termites et les guêpes peuvent construire des nids sophistiqués en coopération, sans que n'importe lequel des individus sache comment procéder (il n'y a pas de plan directeur global). Un autre exemple est le comportement qu'exposent les fourmis ou les abeilles en cherchant la nourriture. Tandis que les fourmis emploient une stratégie de communication indirecte: via la phéromone chimique qu'elles déposent pour trouver les chemins les plus courts entre leur nid et des sources alimentaires, les colonies d'abeilles, quant à elles, sont très efficaces pour l'exploitation des sources alimentaires les plus riches, elles se communiquent des informations sur une nouvelle source de nourriture au moyen d'une pseudo-danse d'agitation [57].

Ainsi, les colonies d'insectes sociaux ont intéressé les chercheurs pendant plusieurs années. Cependant, les principes qui gouvernent leurs comportements sont restés inconnus pendant longtemps [56]. Au cours de ces dernières années, ces principes ont été mis en œuvre avec succès à une série d'applications incluant les algorithmes d'optimisation, les réseaux de communication et la robotique [58].

Comment penser "Swarm Intelligence" ?

L'intelligence en essaim est définie par Bonabeau, Dorigo et Theraulaz comme étant "*Toute tentative visant à concevoir des algorithmes ou bien des dispositifs de résolution de problèmes distribués inspirés par le comportement collectif des insectes sociaux et d'autres sociétés animales*" [66].

Partie II Swarm Intelligence et colonies de fourmis

II.5. Modèles de l'intelligence en essaim

D'un point de vue informatique, les modèles d'intelligence en essaim sont, en grande partie, des algorithmes de recherche stochastiques. Ils sont utiles pour entreprendre des problèmes d'optimisation distribués et multimodaux. Le processus de recherche est robuste et efficace dans le maintien de la diversité. Un mécanisme pour imposer une forme d'oubli est aussi adopté dans certains des algorithmes d'intelligence en essaim de sorte que l'espace de solutions puisse être exploré d'une façon complète.

Ainsi, les algorithmes peuvent éviter la convergence vers une solution locale optimale et, en même temps, parvenir à une solution optimale globale avec une forte probabilité. Le modèle d'intelligence en essaim est simple dans la nature, cependant, le comportement collectif de l'essaim qui émerge des interactions est utile dans la réalisation de buts complexes [59].

L'intelligence en essaim a un caractère pluridisciplinaire marqué, les recherches dans le domaine de l'intelligence en essaim peuvent être classées selon différents critères:

- ✓ *Naturel vs Artificiel*: Il est usuel de diviser la recherche dans le domaine de l'intelligence en essaim en deux secteurs selon la nature des systèmes. On parle donc de recherche en intelligence d'essaim naturelle, où des systèmes biologiques sont étudiés; et d'intelligence d'essaim artificielle, où des artefacts humains sont étudiés.
- ✓ *Scientifique vs Ingénierie*: C'est une classification plus informative de recherche en intelligence en essaim basée sur les buts qui sont poursuivis: on peut identifier un courant scientifique et un courant d'ingénierie. Le but du courant scientifique est de modéliser les systèmes d'intelligence en essaim, ainsi que de choisir et comprendre les mécanismes qui permettent à un système, dans son ensemble, de se comporter d'une façon coordonnée comme résultat à des interactions locales individu-individu et individu-environnement. D'autre part, le but du courant d'ingénierie est d'exploiter la compréhension développée par le courant scientifique pour concevoir les systèmes qui peuvent résoudre les problèmes de pertinence pratique [65].

II.6. Principes de l'intelligence en essaim

Une étude de l'approche "intelligence en essaim" révèle un jeu de principes d'organisation utiles qui peuvent guider la conception d'applications distribuées efficaces pour différentes sortes de problèmes. La *swarm intelligence* a les caractéristiques notables suivantes: [58]

Partie II Swarm Intelligence et colonies de fourmis

- ✓ **Autonomie:** Le système n'exige pas de gestion extérieure ou de maintenance, les individus sont autonomes, contrôlant leur propre comportement d'une manière auto-organisée.
- ✓ **Adaptabilité:** Les interactions entre des individus peuvent surgir par la communication directe ou indirecte via l'environnement local. Deux individus interagissent indirectement quand l'un d'entre eux modifie l'environnement et l'autre répond au nouvel environnement à un temps postérieur. En exploitant de telles formes de communication locales, les individus ont la capacité de détecter des changements au sein de l'environnement dynamiquement, ils peuvent alors adapter leur propre comportement à ces nouveaux changements d'une manière autonome. Ainsi, les systèmes d'essaims présentent des capacités d'auto-configuration.
- ✓ **Evolutivité:** Les capacités de la swarm intelligence peuvent être optimisées en utilisant des groupes comportant un nombre variant de quelques individus jusqu'à des milliers avec la même architecture de contrôle.
- ✓ **Flexibilité:** Aucun individu simple de l'essaim n'est essentiel, c'est-à-dire n'importe quel individu peut être dynamiquement ajouté, enlevé ou remplacé.
- ✓ **Robustesse:** La swarm intelligence fournit un bon exemple d'architecture fortement distribuée, ce qui améliore considérablement sa robustesse. Aucune coordination centrale n'a lieu d'être, ce qui signifie qu'il n'y a aucun point seul d'échec. De plus, comme la plupart des systèmes biologiques et sociaux, et en combinant les capacités d'adaptabilité et flexibilité, le système d'essaim permet la redondance qui est essentielle pour la robustesse.
- ✓ **Parallélisme massif:** Le système d'essaim est massivement parallèle et son fonctionnement est vraiment distribué. Les tâches exécutées par chaque individu dans son groupe sont les mêmes.
- ✓ **Auto-organisation:** Les systèmes d'essaim soulignent des capacités d'auto-organisation. L'intelligence n'est pas présente au niveau de chaque individu, mais apparaît plutôt d'une façon ou d'une autre dans l'essaim entier. Autrement dit, si nous considérons chaque individu comme une unité de traitement, les solutions aux problèmes obtenus ne sont pas prédéterminées ou préprogrammées, mais sont déterminées collectivement suite au programme d'exécution.
- ✓ **Rentabilité:** Les systèmes d'essaim consistent en une collection finie d'agents homogènes, chacun d'entre eux a les mêmes capacités et les mêmes algorithmes de contrôle. Il est clair que l'autonomie et le contrôle fortement distribué, permis par le modèle en essaim, simplifient considérablement la tâche de mise en œuvre d'algorithmes parallèles. Par

Partie II Swarm Intelligence et colonies de fourmis

exemple, pour le type d'essaim des systèmes multi-robotisés, les robots sont relativement simples et leur effort de processus de conception peut être maintenu minimal en termes de capteurs, de déclencheurs et de ressources pour le calcul et la communication.

II.7. Essaims et vie artificielle

Depuis 1990, plusieurs comportements collectifs (comme les insectes sociaux et les masses d'oiseaux) ont inspiré le développement d'algorithmes. Ces algorithmes conviennent bien à des domaines d'application tels que l'étude des problèmes d'optimisation comme des problèmes difficiles tels que: le problème du voyageur de commerce, les problèmes d'affectations quadratique, le regroupement et l'extraction de données ...etc. L'optimisation par essaim de particules (PSO: Particle Swarm Optimization) et l'optimisation par colonies de fourmis (ACO: Ant Colony Optimization) sont actuellement les techniques les plus populaires dans le domaine d'intelligence en essaim. Cependant, on peut distinguer d'autres approches telle que l'optimisation par colonies d'abeilles (BCO: Bee Colony Optimization) [63]. L'avantage de ces approches par rapport aux techniques traditionnelles est leur robustesse et leur flexibilité. Ces propriétés font de l'intelligence en essaim un paradigme de conception intéressant pour traiter des problèmes de plus en plus complexes [55].

II.8. Techniques de la Swarm Intelligence

Dans ce qui suit les principales techniques de la Swarm Intelligences vont être présentées:

II.8.1. L'optimisation par colonies de fourmis (ACO)

La capacité des insectes sociaux à réaliser des tâches hautement complexes a également inspiré les informaticiens et a donné lieux au concept d'intelligence collective articulé autour des mécanismes d'auto-organisation. L'optimisation par colonie de fourmis n'est qu'une des nombreuses applications dans ce domaine [69].

Ainsi, ACO est une métaheuristique qui peut être utilisée pour trouver des solutions approximatives à des problèmes d'optimisations difficiles. Dans ACO, des fourmis artificielles cherchent de bonnes solutions à un problème d'optimisation donné. En appliquant ACO, le problème d'optimisation est transformé en un problème de recherche du meilleur chemin dans un graphe pondéré [71].

J.-L. Deneubourg de l'Université Libre de Bruxelles a proposé une explication aux capacités d'optimisation des procédures de recueil de ressources chez certaines espèces de fourmis. Très

Partie II Swarm Intelligence et colonies de fourmis

schématiquement, au départ, un grand nombre de fourmis se déplacent à l'extérieur du nid, plus ou moins au hasard, à la recherche de nourriture. Quand une fourmi découvre une ressource, elle rentre au nid en déposant une trace de phéromones. Cette trace tend à attirer ses congénères qui, en la suivant, vont parvenir à la nourriture. Ils vont alors retourner au nid et renforcer la trace à leur tour. On assiste ainsi à la mise en place d'une boucle de rétroaction positive. S'il existe plusieurs chemins pour atteindre la ressource, l'accumulation de phéromones sera nécessairement plus rapide sur le chemin le plus court. C'est ainsi que certaines espèces de fourmis tendent à optimiser le recueil de nourriture [72].

Cependant, la phéromone s'évaporera au fil du temps, réduisant donc sa force attractive. Le temps que les fourmis mettent pour traverser le chemin de leur colonie à la source alimentaire et revenir, la phéromone s'évaporera. Autrement dit, la phéromone tout au long du chemin le plus court sera renforcée plus rapidement parce que quand d'autres fourmis suivront le chemin, elles continueront à déposer leur phéromone. En conséquence, le chemin le plus court entre la colonie de fourmis et la source alimentaire apparaîtra finalement. Du point de vue informatique, un des avantages de l'évaporation de phéromone est d'éviter la convergence à une solution localement optimale. En effet, l'évaporation de la phéromone est un mécanisme utile pour imposer une forme d'oubli, permettant à l'espace de solutions d'être exploré d'une façon complète [59].

Marco DORIGO a proposé l'approche connue sous le nom « Optimisation par colonie de fourmis » (*Ant Colony Optimization*) qu'il a d'abord validé en l'appliquant au problème du voyageur de commerce [69]. On trouve maintenant des « fourmis » dans le routage des télécommunications, l'allocation des ressources dans les processus industriels ou encore la conception de circuits électroniques [60].

Les fourmis artificielles sont inspirées du comportement des fourmis réelles, les points de ressemblance entre les deux types de fourmis sont:

- ✓ Les algorithmes de colonies de fourmis sont composés d'une colonie de fourmis artificielles concurrentes et coopératives. Autrement dit les algorithmes de colonies de fourmis se basent sur la notion de population dont les membres (concurrents) coopèrent de manière asynchrone pour trouver de bonnes solutions à un problème donné.
- ✓ Les fourmis artificielles changent quelques informations numériques sauvegardées localement tout comme les fourmis réelles qui déposent des substances chimiques (phéromones). Ces informations constituent les pistes de phéromone artificielle. Elles sont accessibles par toutes les fourmis artificielles. Les pistes de phéromones sont ainsi le seul moyen de communication entre les fourmis.

Partie II Swarm Intelligence et colonies de fourmis

- ✓ Un mécanisme d'évaporation modifie l'information sur les phéromones dans le temps, il permet ainsi de diriger la recherche vers de nouvelles directions.
- ✓ Les fourmis réelles cherchent à minimiser la distance du chemin reliant leurs nids aux sites destinataires. De la même manière, les fourmis artificielles se déplacent d'un état vers un autre état adjacent pour construire une solution complète de moindre coût.

En plus, les fourmis artificielles ont quelques propriétés qui ne se trouvent pas dans les fourmis réelles:

- ✓ Elles ont un état interne privé, représenté par la mémoire des actions du passé.
- ✓ La quantité de phéromones déposée est une fonction de la qualité de la solution trouvée par une fourmi.
- ✓ Les fourmis artificielles peuvent être enrichies par des capacités supplémentaires telles que la prévision, l'optimisation locale ou le retour arrière.
- ✓ Les fourmis artificielles évoluent dans un milieu discret, elles effectuent des déplacements entre des états discrets reliés par des transitions [61].

II.8.2. L'optimisation par colonies d'abeilles (BCO)

On peut considérer une colonie d'insectes sociaux comme un système dynamique rassemblant des informations de l'environnement, il règle son comportement en fonction de ce dernier. Les insectes individuels n'exécutent pas toutes les tâches à cause de leur spécialisation. Généralement, toutes les colonies d'insectes sociaux se comportent selon leur propre répartition des travaux liés à leur morphologie [62].

Dans [62], les chercheurs ont proposé un modèle comportemental d'auto-organisation pour une colonie d'abeilles. Ce modèle a été inspiré par le comportement alimentaire d'abeilles, les abeilles quittent leur colonie pour chercher des sources alimentaires prometteuses. En trouvant une bonne source alimentaire, une abeille retourne à la ruche et informe ses camarades de ruche via une danse d'agitation.

A travers la danse d'agitation d'abeilles qui représente un outil de communication, une abeille transmet trois informations importantes à d'autres abeilles: la distance, la direction et la qualité de la source alimentaire. L'abeille utilise particulièrement cette danse d'agitation pour convaincre d'autres abeilles afin d'être des disciples et retourner à la source alimentaire. En conséquence, plus d'abeilles sont attirées vers des sources alimentaires plus prometteuses. Une telle stratégie est déployée par la colonie d'abeilles pour obtenir une nourriture de qualité d'une façon rapide et efficace [62].

Partie II Swarm Intelligence et colonies de fourmis

Le comportement alimentaire de l'abeille expose des caractéristiques d'auto-organisation, d'adaptabilité et de robustesse. Il y a plusieurs avantages concernant l'optimisation par colonie d'abeilles du point de vue informatique. Parmi ces avantages, nous pouvons citer le fait d'éviter des solutions localement optimales, celui de chercher la meilleure solution basée sur les solutions obtenues par la colonie d'abeilles entière et l'adaptabilité aux changements de l'environnement. Des variantes d'optimisation de colonie d'abeilles ont été proposées, tels que des algorithmes d'abeilles virtuelles et des systèmes d'abeilles flous. Les caractéristiques des abeilles concernant: le comportement alimentaire, l'apprentissage, la mémorisation et les caractéristiques de partage d'informations ont récemment représenté des secteurs de recherche très intéressants dans le domaine de l'intelligence en essaim. Ainsi les études sur les abeilles de miel s'avère être une tendance dans la littérature durant ces dernières années [62].

L'optimisation des colonies d'abeilles a été utilisée pour entreprendre des tâches de transport complexes, d'autres applications incluent le problème de voyageur de commerce, le routage et l'attribution de longueurs d'ondes dans les réseaux optiques [63].

II.8.3. L'optimisation par essaim de particules (PSO)

L'optimisation par essaim de particules est basée sur une technique d'optimisation stochastique. Elle est inspirée par le comportement social des masses d'oiseaux et de bancs de poissons. PSO partage beaucoup de ressemblances avec des techniques de calcul évolutionnaires comme les algorithmes génétiques (AG). Le système est initialisé avec une population de solutions aléatoires et cherche des optimums en mettant à jour des générations. Cependant, à la différence des AG, PSO n'a aucun opérateur d'évolution comme le croisement ou la mutation. Dans PSO, les solutions potentielles, appelées particules, volent au sein de l'espace du problème en suivant les particules optimales actuelles [64].

En optimisation par essaim de particules, des particules cherchent de bonnes solutions à un problème d'optimisation continu. Chaque particule est une solution du problème considéré et utilise sa propre expérience et l'expérience de particules voisines pour choisir la manière de se déplacer dans l'espace de recherche. En pratique, dans la phase d'initialisation on donne une position initiale aléatoire à chaque particule et une vitesse initiale. La position de la particule représente une solution au problème et a donc une valeur, donnée par la fonction objective. En se déplaçant dans l'espace de recherche, les particules retiennent la position de la meilleure solution qu'ils ont trouvée [65].

Partie II Swarm Intelligence et colonies de fourmis

II.9. Applications de la swarm intelligence

Les principes de l'intelligence en essaim ont été appliqués avec succès dans une grande variété de domaines, bien que l'optimisation par colonie de fourmis et l'optimisation par essaim de particules représentent une direction de recherche réussie dans le domaine d'intelligence en essaim, d'autres exemples d'application intéressants peuvent être considérés, tel que la Gestion des réseaux à base d'essaims et le comportement coopératif dans les essaims de robots [58].

II.10. Avantages de la swarm intelligence

L'intelligence en essaim fournit une nouvelle structure pour la conception et la mise en œuvre de systèmes composés de plusieurs agents qui sont capables de coopérer pour résoudre un problème complexe. Les avantages potentiels de l'intelligence en essaim sont variés [65]:

- ✓ *La robustesse collective:* L'échec de composants individuels ne gêne pas significativement la performance;
- ✓ *Le comportement coopératif et la simplicité individuelle:* Permettent de réduire la complexité des individus
- ✓ *L'adaptabilité:* Les mécanismes de contrôle utilisés ne dépendent pas du nombre d'agents dans l'essaim.

L'intelligence en essaim peut donc être vue comme une ingénierie pratique. La majorité des recherches s'est concentrée, jusqu'à présent, sur des démonstrations concernant les capacités des systèmes intelligents à base d'essaim pour la résolution de problèmes nécessitant une coopération. Des résultats très encourageants ont été obtenus, particulièrement dans des applications d'optimisation [65].

Les essaims fournissent une source d'inspiration riche et leurs principes sont directement applicables aux systèmes informatiques. Cependant, bien qu'elle se caractérise par l'auto configuration, l'auto-organisation et la capacité d'adaptation, l'approche en essaim reste utile pour des applications impliquant de nombreuses répétitions de la même activité sur un secteur relativement grand, comme la découverte du chemin le plus court. En effet, l'approche de type "essaim" est basée sur la coopération de grands nombres d'agents homogènes. De telles approches comptent d'habitude sur des résultats de convergence mathématiques qui atteignent le résultat désirable au cours d'une période de temps suffisamment longue. De plus, les agents impliqués sont homogènes [58].

Partie II Swarm Intelligence et colonies de fourmis

II.11. L'approche d'optimisation par colonies de fourmis

L'approche d'optimisation par colonies de fourmis s'est avérée être plus qu'une métaphore amusante présentée aux conférences exotiques: bien que les résultats, obtenus avec le premier algorithme appelé Ant System étaient un peu décevants, des développements récents, qui combinent l'approche de colonie de fourmis avec les méthodes de recherches locales et/ou d'autres méthodes d'optimisation, sont prometteuses [66].

Dans ce qui suit, les concepts fondamentaux sur lesquels cette approche a été bâtie ainsi que sa structure et les différents algorithmes ACO seront présentés:

II.11.1. Inspiration biologique

Le terme "Optimisation par Colonie de Fourmis" a été choisi pour refléter son inspiration originale: le comportement alimentaire d'une certaine espèce de fourmis. Il a été inspiré en particulier par l'expérience de pont à double branche réalisée par *Jean-Louis Deneubourg* et al [72]. Dans cette expérience, il a été démontré que les fourmis peuvent trouver le chemin le plus court à une source alimentaire en exploitant collectivement des phéromones qu'elles déposent sur leurs chemins en se déplaçant.

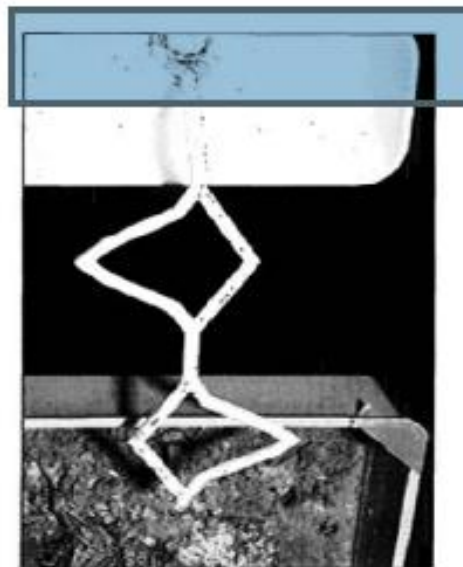


Figure 3 : Illustration de l'expérience réelle du pont à double branche. D'après [72]

Partie II Swarm Intelligence et colonies de fourmis

II.11.2. Théorie de stigmergie

La stigmergie se réfère à la communication indirecte entre les individus d'un système émergent auto-organisé modifiant leur environnement local. La stigmergie est l'un des concepts à la base de la création de la métaheuristique de colonies de fourmis. Elle est précisément définie comme une "forme de communication passant par le biais de modification de l'environnement", mais on peut rencontrer le terme "interactions sociales indirectes" pour décrire le même phénomène. Les biologistes différencient la stigmergie "quantitative" de celle "qualitative", mais le processus en lui-même est identique. La grande force de la stigmergie est que les individus échangent des informations par le biais du travail en cours, de l'état d'avancement de la tâche globale à accomplir [68].

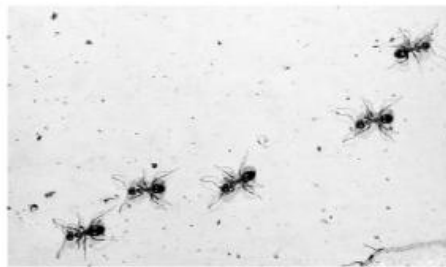


Figure 4 : Des fourmis suivent une piste de phéromone d'après [68]

II.11.3. Auto-organisation et Rétroaction

Le mécanisme permettant de résoudre un problème trop complexe pour être abordé par des fourmis seules est un bon exemple de système auto-organisé. Ce système repose sur des rétroactions positives: le dépôt de phéromone attire d'autres fourmis qui vont la renforcer à leur tour, et des rétroactions négatives: la disparition de la piste par évaporation empêche le système de converger trop rapidement vers une mauvaise solution. Théoriquement, si la quantité de phéromone restait identique au cours du temps sur toutes les branches, aucune piste ne serait choisie. Or, du fait des rétroactions, une faible variation sur une branche va être amplifiée et permettre alors le choix d'une branche [66].

II.11.4. Structure d'ACO

Étant donné un problème d'optimisation combinatoire (CO) qui doit être résolu, en premier on doit tirer un jeu fini C de composants de solution qui sont utilisés pour assembler des solutions du

Partie II Swarm Intelligence et colonies de fourmis

problème de CO. Deuxièmement, on doit définir un jeu de valeurs de phéromone T . Ce jeu de valeurs est généralement appelé le modèle de phéromone, qui est vu, d'un point de vue technique, comme un modèle probabiliste paramétré. Le modèle de phéromone est un des composants centraux d'ACO. Les valeurs de phéromone $\tau_i \in T$ sont d'habitude associées aux composants de la solution. Le modèle de phéromone est utilisé pour générer des solutions au problème étudié en les assemblant à partir du jeu de composants de solution [56].

II.11.5. Métaheuristique ACO

Dans ce qui suit le pseudo code associé à la métaheuristique ACO:

<p>Initialisation des pistes de phéromones;</p> <p>Boucler tant que <i>critère d'arrêt</i> non atteint</p> <p> Construire les solutions composant par composant;</p> <p> Utilisation (facultative) d'une heuristique;</p> <p> Mise à jour des pistes de phéromones;</p> <p>Fin de la boucle.</p>

Figure 5 : Métaheuristique ACO. D'après [69]

Cet algorithme consiste en une étape d'initialisation et une boucle à travers trois composants algorithmiques. Une seule itération de la boucle consiste à construire des solutions par toutes les fourmis, leur amélioration (facultative) avec l'utilisation d'un algorithme de recherche local et une mise à jour des phéromones. Dans ce qui suit, l'explication de ces trois composants algorithmiques plus en détails:

- ✓ *Une construction des solutions*: Un jeu de m fourmis artificielles construit des solutions à partir d'éléments d'un jeu fini de composants de solution disponibles $C = \{c_{ij}\}; i = 1, \dots, n; j = 1, \dots, |D_i|$ (où D_i est l'ensemble de valeurs des variables discrètes). Une construction de solution commence par une solution partielle vide $s^p = \emptyset$; ensuite, à chaque étape de construction, la solution partielle actuelle s^p est étendue en ajoutant un composant de solution faisable à partir du jeu de voisins faisables $N(s^p) \subseteq C$. Le processus de construction de solutions peut être considéré comme un chemin sur le graphe de construction $G_C = (V, E)$. Les chemins permis dans G_C sont implicitement définis par le

Partie II Swarm Intelligence et colonies de fourmis

mécanisme de construction de solution qui définit le jeu N (s^P) avec le respect de la solution partielle s^P [69].

- ✓ *Utilisation (facultative) d'une heuristiques*: Une fois que les solutions ont été construites et avant la mise à jour de phéromones, souvent, certains des actions facultatives peuvent être exigées, elles peuvent être utilisées pour mettre en œuvre le problème spécifique et/ou des actions centralisées, qui ne peuvent pas être exécutées par des fourmis seules. L'action la plus utilisée consiste en l'application de recherche locale aux solutions construites: les solutions localement optimisées sont alors utilisées pour décider quelles phéromones mettre à jour [69].
- ✓ *Mise à jour des pistes de phéromone*: le but de la mise à jour de phéromones est d'augmenter les valeurs de phéromones associées à des solutions prometteuses et ainsi diminuer celles qui sont associées à de mauvaises solutions. D'habitude, ceci est réalisé en diminuant toutes les valeurs de phéromones par l'évaporation de celle-ci et en augmentant les niveaux de phéromones associés à un jeu choisi de bonnes solutions S_{upd} [69].

II.11.6. Algorithmes d'optimisation par colonies de fourmis

L'algorithme d'optimisation par colonies de fourmis est inspiré du comportement des fourmis à la recherche de nourriture. Son principe repose sur le comportement particuliers des fourmis qui, elles sont capable de déterminer le chemin le plus court entre leur nid et une source de nourriture grâce à la phéromone qui est une substance que les fourmis déposent sur le sol lorsqu'elles se déplacent. Lorsqu'une fourmi doit choisir entre deux directions, elle choisit avec une plus grande probabilité celle comportant une plus forte concentration de phéromones.

Fortement inspiré du déplacement des groupes de fourmis, cette méthode a pour but de construire les meilleures solutions à partir des éléments qui ont été explorés par d'autres individus. Chaque fois qu'un individu découvre une solution au problème, bonne ou mauvaise, il enrichit la connaissance collective de la colonie. Ainsi, chaque fois qu'un nouvel individu aura à faire des choix, il pourra s'appuyer sur la connaissance collective pour pondérer ses choix.

L'Algorithme de colonie de fourmis consiste à reformuler le problème à résoudre en un problème de recherche d'un meilleur chemin dans un graphe et à utiliser des fourmis artificielles pour trouver les bons chemins de ce graphe.

A chaque cycle de l'algorithme, chaque fourmi de la colonie construit aléatoirement un chemin du graphe et la quantité de la phéromone est déposée sur les meilleurs chemins découverts lors de ce cycle. Lors des cycles suivants, les fourmis construisent de nouveaux chemins avec une probabilité

Partie II Swarm Intelligence et colonies de fourmis

dépendant de la phéromone déposée lors des cycles précédents et d'une heuristique propre au problème considéré. La colonie de fourmis converge alors peu à peu vers les meilleures solutions. Les algorithmes d'ACO sont essentiellement appliqués à des problèmes combinatoires, notamment du type du voyageur de commerce. Cependant devant le succès rencontré par ces algorithmes, d'autres pistes commencent à être explorées: par exemple l'utilisation de ces algorithmes dans des problèmes continus et/ou dynamiques ou encore l'exploitation de ce type d'algorithmes dans un cadre d'intelligence en essaim [68].

Plusieurs cas particuliers de la métaheuristique ACO ont été proposés dans la littérature, dans ce qui suit, nous présentons brièvement une vue d'ensemble de trois algorithmes: Ant system (AS), Ant Colony System (ACS) et MAX-MIN Ant System (MMAS).

✓ **Ant system (AS):** Il vise notamment à résoudre le problème du voyageur de commerce, (TSP: Travelling Salesman Problem) où le but est de trouver le plus court chemin permettant de relier un ensemble de villes. L'algorithme général est relativement simple, et repose sur un ensemble de fourmis, chacune parcourant un trajet parmi ceux possibles. À chaque étape, la fourmi choisit de passer d'une ville à une autre en fonction de quelques règles:

- elle ne peut visiter qu'une fois chaque ville;
- plus une ville est loin, moins elle a de chance d'être choisie (c'est la « visibilité »);
- plus l'intensité de la piste de phéromone disposée sur l'arête entre deux villes est grande, plus le trajet aura de chance d'être choisi;
- une fois son trajet terminé, la fourmi dépose, sur l'ensemble des arêtes parcourues, plus de phéromones si le trajet est court;
- les pistes de phéromones s'évaporent à chaque itération.

Le problème est plus généralement défini comme un graphe complètement connecté $G(V, E)$ où les villes sont les sommets V et les trajets entre les villes, les arêtes E .

A tout instant t , chaque fourmi choisit une ville de destination, elles se déplacent à l'instant $t+1$ vers une ville de son choix. On appelle une itération de l'algorithme AS, l'ensemble des déplacements de l'ensemble de la colonie entre l'instant t et l'instant $t+1$. Ainsi après n itérations, l'ensemble de la colonie aura effectué un circuit hamiltonien (un circuit qui passe exactement une fois par tous les sommets du graphe) sur le graphe. De cette manière, toutes les fourmis commenceront et finiront leur tour en même temps.

Partie II Swarm Intelligence et colonies de fourmis

Choix des transitions: Une fourmi k placée sur la ville i à l'instant t va choisir sa ville j de destination en fonction de la visibilité μ_{ij} de cette ville et de la quantité de phéromones $\tau_{ij}(t)$ déposée sur l'arc reliant ces deux villes. Ce choix sera réalisé de manière aléatoire, avec une probabilité de choisir la ville j donnée par :

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot \eta_{ij}^\beta}{\sum_{l \in N_i^k(t)} [\tau_{il}(t)]^\alpha \cdot \eta_{il}^\beta} & \text{si } j \in N_i^k \\ 0 & \text{sinon} \end{cases} \quad (12)$$

Où: N_i^k étant l'ensemble des villes que la fourmi k , placée sur la ville i , n'a pas encore visité à l'instant t dans le cycle courant, η_{ij} la visibilité, qui est égale à l'inverse de la distance de deux villes i et j ($1/d_{ij}$) et $\tau_{ij}(t)$ l'intensité de la piste à une itération donnée t . α et β sont deux paramètres qui contrôlent l'importance relative entre phéromones et visibilités. Ainsi si α est égal à 0, le choix se fera uniquement en fonction de la visibilité (si β est différent de zéro) [70].

Mise à jour des phéromones: Une fois la tournée des villes effectuée, une fourmi k dépose une quantité ΔT_{ij}^k de phéromones sur chaque arête de son parcours:

$$\Delta T_{ij}^k(t) = \begin{cases} \frac{Q}{L^k(t)} & \text{si } (i, j) \in T^k(t) \\ 0 & \text{sinon} \end{cases} \quad (13)$$

Où $T^k(t)$ est la tournée faite par la fourmi k à l'itération t , $L^k(t)$ la longueur du trajet et Q un paramètre de réglage.

A la fin de l'itération, on a la somme des phéromones qui ne se sont pas évaporées et de celles qui viennent d'être déposées:

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^m \Delta T_{ij}^k(t) \quad (14)$$

$\rho\tau_{ij}(t)$: représente l'évaporation des phéromones déposées aux itérations précédentes par les fourmis, m est le nombre de fourmis utilisées pour l'itération t et ρ un paramètre de réglage, $\rho \in [0, 1[$. Il définira la vitesse d'évaporation des phéromones sur les arcs entre l'instant t et l'instant $t+n$. Le choix de ρ est important, en effet, si ρ se rapproche trop de 1, on observe un effet de stagnation des phéromones sur les arcs, ce qui implique des inconvénients tel que le fait de voir les mauvaises solutions persister. De même, choisir $\rho \approx 0$ implique une évaporation trop rapide des phéromones, ce qui amène donc la fourmi à un choix dépendant uniquement de la visibilité des nœuds [73].

Partie II Swarm Intelligence et colonies de fourmis

- ✓ **Ant Colony System (ACS):** La contribution la plus intéressante d'ACS est l'introduction d'une mise à jour locale de phéromones (la mise à jour pas à pas effectué sur un arc lorsqu'une fourmi l'emprunte), en plus de la mise à jour globale de phéromones exécutée à la fin du processus de construction. ACS a tendance de diminuer la teneur en phéromones. A l'inverse d'AS, ACS augmente lors de la mise à jour globale (après chaque cycle) uniquement le chemin appartenant au meilleur tour (celui de longueur minimale). La mise à jour locale de phéromones est exécutée par toutes les fourmis après chaque étape de construction. Chaque fourmi l'applique seulement au dernier bord traversé comme suit: [74]

$$\tau_{ij} = (1 - \varphi) \cdot \tau_{ij} + \varphi \cdot \tau_0 \quad (15)$$

Ou $\varphi \in [0, 1]$ est le coefficient de décrépitude de phéromones et τ_0 est la valeur initiale de la phéromone. Le but principal de la mise à jour locale est de diversifier la recherche exécutée par des fourmis ultérieures pendant une itération en diminuant la concentration de phéromones sur les arcs traversés, les fourmis encouragent des fourmis ultérieures à choisir d'autres arcs et, de là, produire des solutions différentes, et ainsi il y aura une faible probabilité que plusieurs fourmis produisent des solutions identiques pendant une itération. La mise à jour globale de la phéromone est appliquée à la fin de chaque itération par une seule fourmi seulement. La formule de mise à jour est légèrement différente: [74]

$$\tau_{ij} \leftarrow \begin{cases} (1 - \rho) \cdot \tau_{ij} + \rho \cdot \Delta\tau_{ij} & \text{si } (i, j) \text{ appartient au meilleur tour} \\ \tau_{ij} & \text{sinon.} \end{cases} \quad (16)$$

- ✓ **MAX MIN Ant System (MMAS):**

MMAS est fondé sur l'algorithme AS, néanmoins il présente quelques différences notables:

1. Seule la meilleure fourmi met à jour une piste de phéromones.
2. Les valeurs des pistes sont bornées par τ_{\min} et τ_{\max} .
3. Les pistes sont initialisées à la valeur maximum τ_{\max} .
4. La mise à jour des pistes se fait de façon proportionnelle, les pistes les plus fortes étant moins renforcées que les plus faibles.
5. Une réinitialisation des pistes peut être effectuée.

Les meilleurs résultats sont obtenus en mettant à jour la meilleure solution avec une fréquence de plus en plus forte au cours de l'exécution de l'algorithme.

L'équation de la mise à jour de la phéromone prend la forme suivante: [75]

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \Delta\tau_{ij}^{best} \quad (17)$$

Partie II Swarm Intelligence et colonies de fourmis

Où $\Delta\tau_{ij}^{best}$ est la valeur de mise à jour de la phéromone définie par:

$$\Delta\tau_{ij}^{best} = \begin{cases} \frac{Q}{L_{best}} & \text{si la meilleure fourmi utilise l'arête } (i,j) \text{ dans son tour} \\ 0 & \text{sinon} \end{cases} \quad (18)$$

Où L_{best} est la longueur du tour effectué par la meilleure fourmi [75].

Les valeurs de phéromones sont contraintes entre τ_{min} et τ_{max} en vérifiant, après qu'elles soient mises à jour par les fourmis, que toutes les valeurs de phéromones sont dans les limites imposées: τ_{ij} est posé à τ_{max} si $\tau_{ij} > \tau_{max}$ et à τ_{min} si $\tau_{ij} < \tau_{min}$.

Il est important de noter que l'équation de mise à jour de phéromones de MMAS est appliquée, comme c'est le cas pour AS, à tous les arcs tandis que dans ACS est appliquée seulement aux arcs visités par les meilleures fourmis.

La valeur minimale est plus souvent expérimentalement choisie, la valeur maximale quand elle peut être calculée analytiquement à condition que l'on connaisse la longueur de tour optimal de la fourmi [75].

II.11.7. Exemple d'utilisation d'ACO dans un problème de classification de textes

La majorité des travaux utilisant ACO pour la classification de textes sont consacrés à la classification non supervisée, clustering [76, 77, 78].

Dans une méthode de classification non supervisée (clustering) de textes, aucune information sur la classe d'un texte n'est fournie à la méthode (on dit que les textes ne sont pas étiquetés). En classification supervisée, les textes sont étiquetés tout en connaissant leurs dissimilarités. Le problème consiste alors à construire des hyper plans séparant les textes selon leur classe. L'objectif de la classification non supervisée est opposé au cas de la classification supervisée: dans le premier cas, il s'agit de découvrir des groupes de textes alors que dans le deuxième il s'agit d'utiliser les groupes connus pour découvrir ce qui les différencie ou afin de pouvoir classer de nouveaux textes dont la classe n'est pas connue. Dans ce qui suit, nous présentons deux travaux de classification non supervisée de documents textuels.

- Julia Handl et al. dans [76], proposent un système de recherche de documents pour le moteur de recherche sur internet, basé sur une variante de l'algorithme de Lumer et Faieta [79]. L'algorithme permet de résoudre la classification des documents renvoyés par un moteur de recherche et de produire en même temps une carte des résultats, à partir des tas d'objet répartis sur la grille par les fourmis artificielles. La carte résultante est appelée *carte des centres d'intérêts* ou *topic map* et telle que les documents similaires sont affichés au même endroit et qu'inversement, les documents sans rapport ont des localisations

Partie II Swarm Intelligence et colonies de fourmis

différentes. Les poids des termes de documents sont déterminés par la méthode de pondération TF-IDF. Les auteurs ont adopté une méthode qui permet de mettre en place une mémoire à court terme et ils ont employé des fourmis avec des vitesses différentes, leur permettant de réaliser des sauts ou jumps. En outre, ils ont introduit une stratégie adaptative de mise à l'échelle, ainsi que quelques autres modifications pour obtenir des résultats fiables.

- Ramos et Marcelo [77] ont développé une nouvelle stratégie appelée *ACLU* pour aborder la classification non supervisée de documents textuels. Les auteurs ont proposé l'utilisation des probabilités de transition bio-inspirés, en évitant le déplacement aléatoire des fourmis, qui peuvent explorer des régions non-intéressantes. Les fourmis se déplacent selon les probabilités de transition qui dépendent de la répartition spatiale de la phéromone dans l'environnement. Si un groupe particulier disparaît, la phéromone a tendance à s'évaporer de cet emplacement. Cette approche est intéressante, parce que la phéromone représente la mémoire des essaims et toutes les fourmis peuvent en bénéficier. En d'autres termes, les fourmis partagent une mémoire commune.

Très peu de travaux abordent la classification supervisée de documents textuel en utilisant les colonies de fourmis [80, 81].

Dans [81], les auteurs traitent la classification des documents textuels dans un papier intitulé « Text Document Classification using Ant Colony Optimization and Genetic Algorithm ». Ils se basent dans leur conception sur le travail de [80] pour représenter le graphe de documents ainsi que l'adaptation de l'algorithme d'optimisation de colonies de fourmis pour la classification de documents.

Dans ce qui suit, on expose le travail Lachtar et Bahi:

- Lachtar et Bahi dans [80] utilisent l'algorithme de colonie de fourmis pour classer des documents textuels. Un graphe de documents est construit. Chaque nœud du graphe est un document, les arcs sont valués par la similarité cosinusoidale entre document qui représente la phéromone. Pour classer un nouveau document, chaque fourmis parte d'un nœud initial et explore le graphe à la recherche du document le plus similaire avec le document à classer. La transition de la fourmi k du nœud i vers un nœud j est calculée par la formule suivante :

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha * [s_{ij}(t)]^\beta}{\sum_{k \in Z_k} [\tau_{ij}(t)]^\alpha * [s_{ij}(t)]^\beta} \quad (19)$$

Partie II Swarm Intelligence et colonies de fourmis

Z_k représente la liste des documents non encore visités par la fourmi k

$\tau_{ij}(t)$: représente la quantité de phéromones entre le document i et le document j

s_{ij} : est la similarité cosinusoidale entre le document i et le document j donné par la formule suivante:

$$\text{cosine}(d_i, d_j) = \frac{\sum_{k=1}^n (w_{ik} * w_{kj})}{\sqrt{\sum_{k=1}^n (w_{ik})^2 * \sum_{k=1}^n (w_{kj})^2}} \quad (20)$$

α Et β sont deux paramètres qui contrôlent l'importance relative entre phéromones et visibilité entre documents (similarité cosinusoidale)

La mise à jour de la phéromone est donnée par la formule suivante

$$\tau_{ij}(t + 1) = (1 - \rho) * \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (21)$$

$$\Delta\tau_{ij}(t) = \begin{cases} \frac{n}{L_k(t)} & \text{si } i, j \in T^k(t) \\ 0 & \text{sinon} \end{cases} \quad (22)$$

$T^k(t)$ est l'ensemble de documents qui appartiennent au chemin construit par la fourmi k .

$L_k(t)$ est la longueur du chemin construit par la fourmi k , c'est la somme de la similarités cosinusoidale des document appartenant au chemin, n : est le nombre des documents.

II.12. Conclusion

L'intelligence en essaim représente une nouvelle manière de contrôler les systèmes impliquant plusieurs agents. Cette approche, ainsi que la stratégie émergente sont basées sur un grand nombre d'agents homogènes, chacun d'entre eux possède des capacités assez limitées. Le comportement peut apparaître suite aux interactions locales des agents et les interactions entre les agents et l'environnement. Une question de recherche clé dans un tel scénario détermine la conception appropriée des lois de contrôle locales qui permettront à la collection d'agents de résoudre un problème donné. L'approche d'optimisation par colonies de fourmis est une branche de l'intelligence en essaim. Bien que l'algorithme de colonie de fourmis soit conçu au départ pour résoudre le problème du voyageur de commerce, il offre finalement beaucoup de souplesse pour résoudre avec succès d'autres problèmes complexes tels que la catégorisation de textes. Dans la partie suivante de ce chapitre, nous adaptons l'algorithme de colonies de fourmis pour la catégorisation de chansons. Notre choix est motivé par la flexibilité de cette métaheuristique qui rend possible son application à différents problèmes qui ont pour point commun d'être NP-difficile. Ainsi l'utilisation d'un modèle parallèle (colonies des fourmis) diminuent le temps de calcul et améliorent la qualité des solutions obtenues pour la catégorisation.

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

III. Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

III.1. Introduction

Dans cette partie, nous proposons une approche originale qui réside dans le fait d'adapter un algorithme de colonies de fourmis pour identifier le thème d'une chanson afin de la classer.

Un algorithme de colonies de fourmis est un algorithme itératif à population où tous les individus partagent un savoir commun qui leur permet de guider leurs futurs choix et d'indiquer aux autres individus des directions à suivre ou au contraire à éviter. Il est défini dans le chapitre 2, partie 2, section 11.6. Dans ce qui suit, on expose l'essentiel de l'algorithme de colonies de fourmis, l'algorithme repose sur le comportement particulier des fourmis et permet de déterminer le chemin le plus court entre le nid et une source de nourriture. Les itérations de l'algorithme correspondent aux déplacements des fourmis. Pour aller d'un nœud du graphe à un autre, chaque fourmi aura besoin d'un nombre d'itérations dépendant de la taille de l'arc de graphe à parcourir. Ce mode d'itération va aussi privilégier les plus courts chemins puisque les fourmis auront besoin de moins d'itérations pour en arriver au bout. Chaque fourmi doit connaître la liste des nœuds qu'elle a déjà parcourus et les nœuds encore à parcourir. De plus elle doit mesurer le temps qu'elle passe sur la solution qu'elle explore. À chaque nœud, la fourmi va étudier les arcs possibles en observant leurs niveaux de phéromones respectifs. Elle n'a ensuite qu'à choisir au hasard, en privilégiant les arcs fortement phéromonés. Une fois arrivée à destination, la fourmi connaît la longueur totale de la solution qu'elle a trouvée, elle peut refaire le chemin en sens inverse pour marquer le chemin avec ses phéromones et enrichir la connaissance collective de la colonie. Le moteur de l'algorithme est un conteneur de fourmis qui déroule les itérations. Le problème contient le graphe à explorer, ainsi que les phéromones de chaque arc. Il contient la connaissance collective de la colonie, sous forme d'une matrice de phéromones [41].

III.2. Ressemblance de la catégorisation de chanson avec le problème de voyageur de commerce (TSP)

La ressemblance entre le problème de TSP et la classification de chansons basée sur un algorithme ACO est la suivante:

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

- 1- Une classe de fourmis portant le même nom de catégories correspond à une catégorie cible qui est déterminée par un mécanisme de classification.
- 2- La distance entre les villes peut être considérée comme la similarité entre les catégories et les nœuds.
- 3- Les phéromones correspondent à des poids de classification des nœuds.

III.3. Le processus d'apprentissage

Les catégories des chansons sont déterminées à l'avance. Le poids w_{kj} du terme j de la catégorie k peut être calculé par la formule suivante:

$$w_{kj} = \frac{tf_i * \log\left(\frac{N}{n_i}\right)}{\sqrt{\sum_i (tf_i^2) * \log^2\left(\frac{N}{n_i}\right)}} \quad (23)$$

Où N est le nombre total de chansons, tf_i est la fréquence du terme i et n_i est la fréquence de chansons contenant le terme i (le nombre total de chansons d'apprentissage incluant le terme i).

Chaque catégorie k sera représentée par un vecteur contenant le terme i et son poids w_{ik} . A la fin on ajoute au vecteur le genre de musique dominant de cette catégorie.

$$g_c = \max_{gf_i} = \text{Max}_i \frac{nb}{N} \quad (24)$$

Où N est le nombre total de chansons, nb est le nombre de chansons du corpus d'apprentissage appartenant au genre i

On utilise le genre de musique pour prendre en considération les caractéristiques de l'artiste lors de la classification de chansons puisque chaque artiste a tendance à composer un genre bien déterminé de musique et le genre de musiques a une relation étroite avec le thème de chansons.

La similarité cosinusoidale permet de comparer des chansons et des catégories de longueurs différentes en normalisant leurs vecteurs et elle met l'accent plutôt sur la présence de mots que sur l'absence de mots (la présence de mots est probablement plus représentative de la catégorie de la chanson que l'absence de mots).

Chaque catégorie k sera représentée par un vecteur contenant le terme i et son poids w_{ik} et g_c le genre de musique dominant.

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

III.4. Construction du graphe de chansons

Nous considérons un ensemble de chansons, une mesure de similarité entre ces chansons. Dans notre travail, on considère que chaque nœud du graphe est un terme d'une chanson et les liens sont créés par les mesures de similarités entre les termes des chansons. Le graphe de chansons est donné par la figure 6.

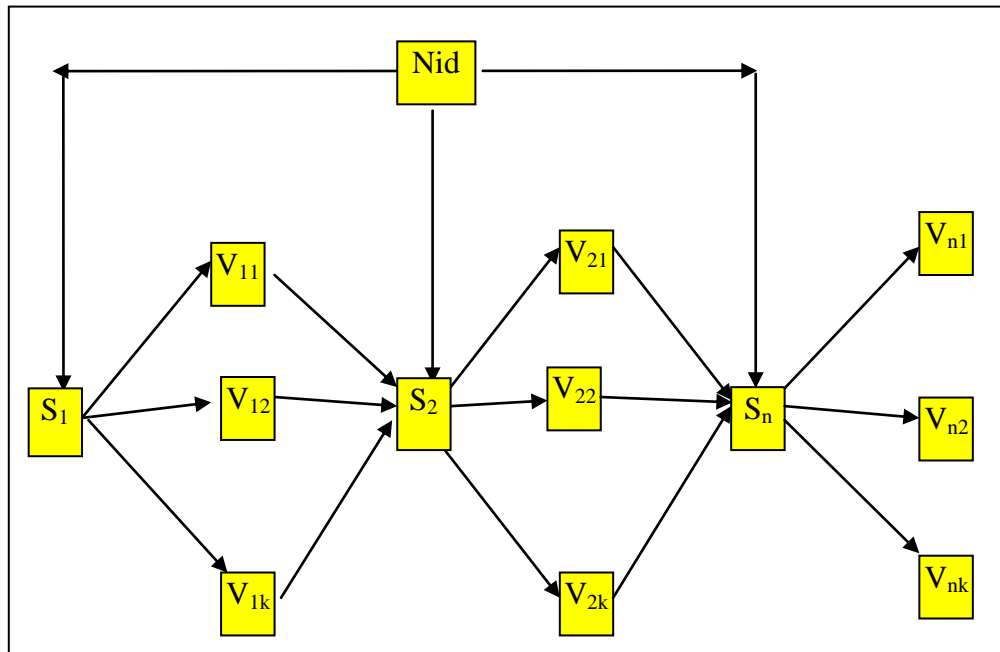


Figure 6: Représentation du graphe de chansons Chans-Graph [41]

Nœud = chanson S = enregistrement contenant un vecteur de caractéristiques V (ensemble de termes).

Arc = mesure de similarité entre deux termes. Les arcs du graphe sont étiquetés avec les traces de phéromones initiales.

Les graphes sont considérés comme d'excellents outils permettant la représentation de données structurées. L'intérêt de les utiliser dans notre processus de catégorisation se trouve certainement dans leur puissance à modéliser les dépendances entre les chansons.

Ainsi, l'idée est d'appliquer un modèle de calcul de similarité entre la chanson à classer et les catégories. Ce modèle se réduit à la comparaison des catégories et le graphe de chansons. D'un point de vue mathématique, le calcul de similarité est réalisé par une recherche de morphisme de graphes. Ce genre de problème s'appelle appariement de graphes qui est un problème NP-complet.

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

Dans le processus de catégorisation, nous nous intéressons à la recherche de la meilleure catégorie qui décrit le thème de la chanson à classer. Cette recherche peut se traduire en un problème de sélection de sous-ensembles (SS-problème), dans le but de trouver un sous-ensemble qui satisfait certaines propriétés. Pour résoudre ce problème, nous utilisons une méthode d'optimisation par métaheuristique. Les métaheuristicques sont des méthodes approchées qui traitent les problèmes d'optimisation difficile. Nous proposons d'utiliser l'algorithme d'optimisation par les colonies de fourmis qui est une métaheuristique récente qui s'inspire de l'intelligence collective des fourmis. Notre choix est motivé par la flexibilité de cette métaheuristique qui rend possible son application à différents problèmes qui ont pour point commun d'être NP-difficile. Ainsi l'utilisation d'un modèle parallèle (colonies des fourmis) diminuent le temps de calcul et améliorent la qualité des solutions obtenues pour la catégorisation.

Dans notre travail, l'algorithme d'optimisation par colonies de fourmis est paramétré par un ensemble de caractéristiques pour être capable de retourner le meilleur chemin. Le chemin avec la concentration de phéromone la plus élevée. La catégorie de ce chemin est la catégorie de la chanson.

III.5. Calcul de distances entre la chanson à classer et les chansons constituant le graphe

Pour notre approche on utilise la similarité cosinusoidale entre une chanson a_i et une catégorie c_j définie par

$$s(a_i, c_j) = \text{cosine}(a_i, c_j) = \frac{\sum_{k=1}^n (w_{ik} * w_{kj})}{\sqrt{\sum_{k=1}^n (w_{ik})^2 * \sum_{k=1}^n (w_{kj})^2}} \quad (25)$$

Où n est l'ensemble des attributs.

L'algorithme suivant calcule la similarité cosinusoidale basée sur les attributs pertinents pour les différents couples constituant les nœuds du graphe de chansons et les catégories. Il prend en entrée le graphe de chansons et les catégories et retourne en sortie une matrice de similarité.

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

Algorithme_Calcul_Similarité_cosinusoidale

```
Entrée : Chans_Graph, Catégories //graphe de chansons, les catégories;
Sortie : Mat_Sim // matrice de similarité
Mat_Sim ← 0 ;
Début
/* Extraire l'ensemble d'attributs des nœuds du graphe
Pour chaque nœud1 du Chans_Graph faire
/* Extraire l'ensemble d'attributs des nœuds de la catégorie
Pour chaque nœud2 des catégories
SIM= Calcul_Sim (Nœud1, Nœud2) ;
Mat_Sim = Mat_Sim + Sim (Nœud1, Nœud2);
Return Mat_Sim
Fin.
```

III.6. Construction d'une solution

Formalisation du problème. Dans notre contexte, le problème de classification d'une nouvelle chanson se ramène au problème de sélection d'un sous-ensemble que nous pouvons le formaliser par le couple (S, f) tel que:

- S contient l'ensemble des couples liant un nœud du graphe Chans_Graph avec un nœud des catégories par la relation de similarité. C'est « la matrice de similarité » Mat_sim.
- f est définie par la fonction score, on s'inspire du travail présenté dans [82], pour déterminer la fonction score, définie par la formule suivante :

$$score(s') = f(\text{Chans}_{\text{Graph}} \cap \text{catégorie})$$

Donc, le résultat est un sous-ensemble de couples de nœuds $S' \in S$, tel que la fonction score soit maximale.

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

Les fourmis artificielles construisent une solution en se déplaçant d'un terme à un autre sur le graphe voir figure 7.

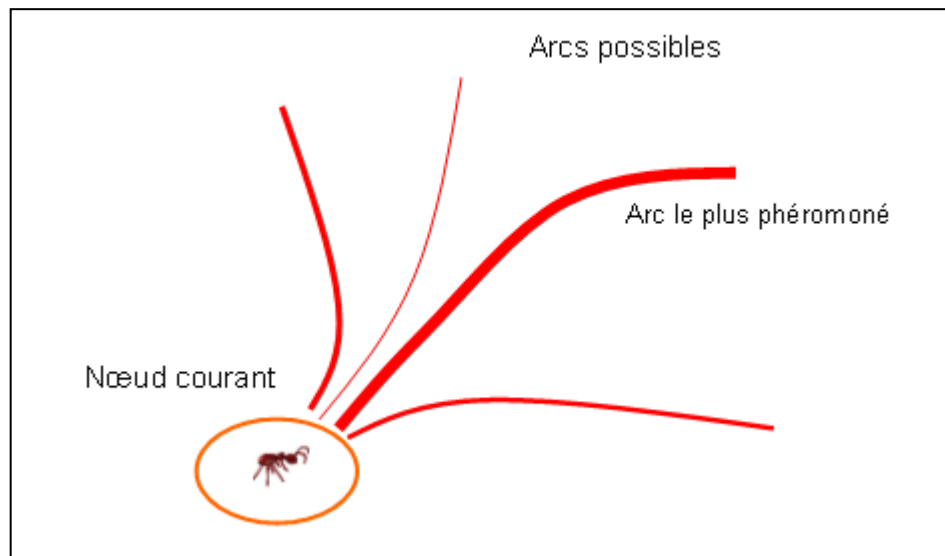


Figure 7 : Déplacement d'une fourmi

Pour passer d'un terme à l'autre, une fourmi doit analyser les choix possibles. Le choix des termes suivants en tenant compte de la connaissance collective.

III.7. Description de l'algorithme

Toutes les fourmis sont divisées en plusieurs groupes (le nombre de groupes est égale au nombre de catégories).

A chaque cycle de l'algorithme, chaque groupe de fourmis construit un sous-ensemble de nœuds couverts. En partant d'un sous-ensemble I_k vide, les fourmis ajoutent à chaque itération un nœud à I_k choisi parmi l'ensemble des nœuds non encore sélectionnés.

Les fourmis de chaque groupe traversent tous les nœuds du graphe. Le nombre de fourmis dans un groupe détermine le nombre d'itérations de ce groupe.

Un chemin I_k de la classe k qui peut décrire l'optimum de cette classe sera généré après que les fourmis du groupe k ont visité tous les nœuds. Le résultat de classification sortira en comparant les concentrations de phéromone b des chemins I_k après toutes les itérations des fourmis de tous les groupes. La classification k décrite par le chemin I_k qui a la concentration de phéromone max est la classe de la chanson.

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

a- Détermination du prochain nœud du chemin

Le nœud suivant du chemin est déterminé par la similarité et la probabilité de transition du nœud courant. La similarité d'un nœud peut être calculée en utilisant la formule (25)

$$\text{cosine}(a_i, c_j) = \frac{\sum_{k=1}^n (w_{ik} * w_{kj})}{\sqrt{\sum_{k=1}^n (w_{ik})^2 * \sum_{k=1}^n (w_{kj})^2}} \quad (25)$$

La probabilité de transition peut être calculée en utilisant la formule comme suit:

$$P_{ij} = \frac{\tau_j}{\sum_{\tau} \tau_i} \quad (26)$$

b- Calcul de la phéromone τ_j à mettre à jour après obtention d'un nœud j

$$\tau_j = \rho \tau_i + \Delta \tau_j \quad (27)$$

$\Delta \tau_j$ est égal à w_{jk} qui est le poids du terme j appartenant à la catégorie k.

c- Trouver le chemin optimal couvrant la collection dans chacune des catégories:

La similarité de chaque chemin I_k (la catégorie de la chanson) peut être calculée par la concentration de phéromone. La formule ci-dessous donne le calcul de phéromone.

$$b_k = \frac{\sum_{i=1}^n \tau_i}{n} \quad (28)$$

Construction d'une solution. Le code suivant décrit la procédure suivie par les fourmis pour construire un sous-ensemble. Le premier objet est choisi aléatoirement, les objets suivants sont choisis au sein de l'ensemble nœuds.

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

Procédure-construction-solution

Entrée: Un SS-problème (S, f) et une fonction heuristique associée: $S^*P(S) \rightarrow \mathbb{R}_+$;

Un facteur phéromonal τ , le nombre de fourmis par groupe, M le nombre de catégories, une valeur standard

Sortie: Une classe C

Début

Divisez toutes les fourmis en m groupes (m est le nombre de catégories).

Les termes des chansons sont hachés au hasard.

Pour k allant de 1 à m

Chaque fourmi du groupe de la classe k (a_k) traverse tous les nœuds.

Tous les nœuds sont initialisés avec la même valeur de phéromone τ_0 .

Sélectionnez un nœud de départ au hasard et commencer l'exploration après avoir calculé la phéromone τ_1 .

L'ensemble de nœuds couvert est initialisé à un ensemble vide, $I = \{ \}$.

Tant que (il y'a une fourmi du groupe non engagé)

Le prochain nœud j est choisi en fonction de la valeur maximale de x qui est le produit de similarité avec le nœud courant i et la probabilité de transition

$x = s * p_{ij}$, les valeurs de s et p_{ij} peuvent être calculées respectivement par les formules 25 et 26

Si ($x <$ valeur standard) alors l'exploration de cette fourmi est terminée

Sinon

Accès au nœud j et mise à jour de phéromones par la formule (27)

$I = I \cup \{j\}$ (Éliminer les nœuds redondants)

Mettre à jour la collection de couverture I

Fin si

Fin tant que

Obtenir la collection couvrant la classe k , $I_k = \{I_1, I_2, \dots, I_n\}$

Calcul de la concentration de phéromone en utilisant la formule (28)

La prochaine k

Fin Pour

La catégorie de la chanson est donnée par $\text{Max}(b_k)$.

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

III.8. Expérimentations et discussions

Pour tester notre algorithme, nous avons effectué des expériences sur deux corpus les plus variés possibles (un pour l'ensemble d'apprentissage sur lequel on peut évaluer les performances et l'autre pour l'ensemble de test qui contient des chansons dont on connaît à l'avance les catégories auxquelles elles devaient appartenir). Nous avons utilisé le classificateur Bayes naïf présenté dans la section 3 partie 1 chapitre 2 comme un classificateur de référence.

125 chansons sont sélectionnées, 67 représentent le corpus d'apprentissage incluant 12 chansons sur l'hymne nationaux, 30 sur l'amour, 13 sur le chants religieux, 4 sur le sport et 8 sur les chansons éducatives et les 58 chansons restantes représentent le corpus de test.

234 termes sont extraits, les poids des termes des classes sont exprimés en W et sont calculés dans le processus d'apprentissage selon la formule (23).

Chaque chanson du test est classée par un calcul itératif dans le processus d'apprentissage en utilisant l'algorithme de classification ci-dessus.

III.8.1. Résultats

Les résultats de classification sont évalués par le taux de précision et le taux de rappel.

Le nombre de fourmis pour chaque groupe et la valeur standard sont respectivement 200 et 0.70.

Le nombre de groupe de fourmis est égal au nombre de catégorie ici cinq.

$\tau_0 = 5$ et $\rho = 0,5$, Les valeurs de ces paramètres sont déterminées à travers des expériences effectuées dans [83].

Le tableau 1 détaille les classes ainsi que le nombre de chansons dans le corpus d'apprentissage et le corpus de test pour chaque catégorie.

Tableau 1: Classes du corpus

Classes	Nombre de chansons dans le corpus d'apprentissage	Nombre de chansons dans le corpus de test
Hymnes nationaux	12	10
Chansons d'amours	30	26
Chants Religieux	13	12
Chansons Sportives	4	4
Chansons Educatives	8	6

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

Les résultats des classifications sont reportés dans les tableaux ci dessous pour le classificateur Bayes Naïf (tableau 2) et l'algorithme de colonie de fourmis (tableau 3).

Tableau 2: Résultats des tests avec l'algorithme des colonies de fourmis

Classes	Hymnes Nationaux	Chansons d'Amours	Chants Religieux	Chansons Sportives	Chansons Educatives	Total
Hymnes Nationaux	9	0	1	0	0	10
Chansons d'Amours	0	24	1	0	1	26
Chants Religieux	0	3	8	0	1	12
Chansons Sportives	0	1	0	3	0	04
Chansons Educatives	0	1	0	0	5	06

Tableau 3: Resultats des tests avec l'algorithme Bayes naïf

Classes	Hymnes Nationaux	Chansons d'Amours	Chants Religieux	Chansons Sportives	Chansons Educatives	Total
Hymnes Nationaux	7	2	1	0	0	10
Chansons d'Amours	0	22	1	1	2	26
Chants Religieux	1	3	7	0	1	12
Chansons Sportives	0	1	0	3	0	04
Chansons Educatives	1	1	0	0	4	06

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

Pour évaluer notre approche, On utilise les mesures de performances définies dans la section 4 partie I de ce chapitre

Le tableau 4 et le tableau 5 présentent les performances de l'algorithme de colonie de fourmis et le classificateur Bayes naïf en terme de rappel, de précision et F1 pour chaque classe.

Tableau 4: Rappel, Précision et F1 pour chaque classe (colonie de fourmis)

Classes	Rappel r	Précision p	F1
Hymnes Nationaux	90,00	100	94,73
Chansons d'Amours	92,30	82,75	87,26
Chants Religieux	66,66	80	72,72
Chansons Sportives	75,00	100	85,71
Chansons Educatives	83,33	71,42	76,91

Tableau 5 : Rappel, Précision et F1 pour chaque classe (Bayes naïf)

Classes	Rappel r	Précision p	F1
Hymnes Nationaux	70,00	77,77	73,68
Chansons d'Amours	84,61	75,86	79,99
Chants Religieux	58,33	77,77	66,66
Chansons Sportives	75,00	75,00	75,00
Chansons Educatives	66,66	57,14	61,53

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

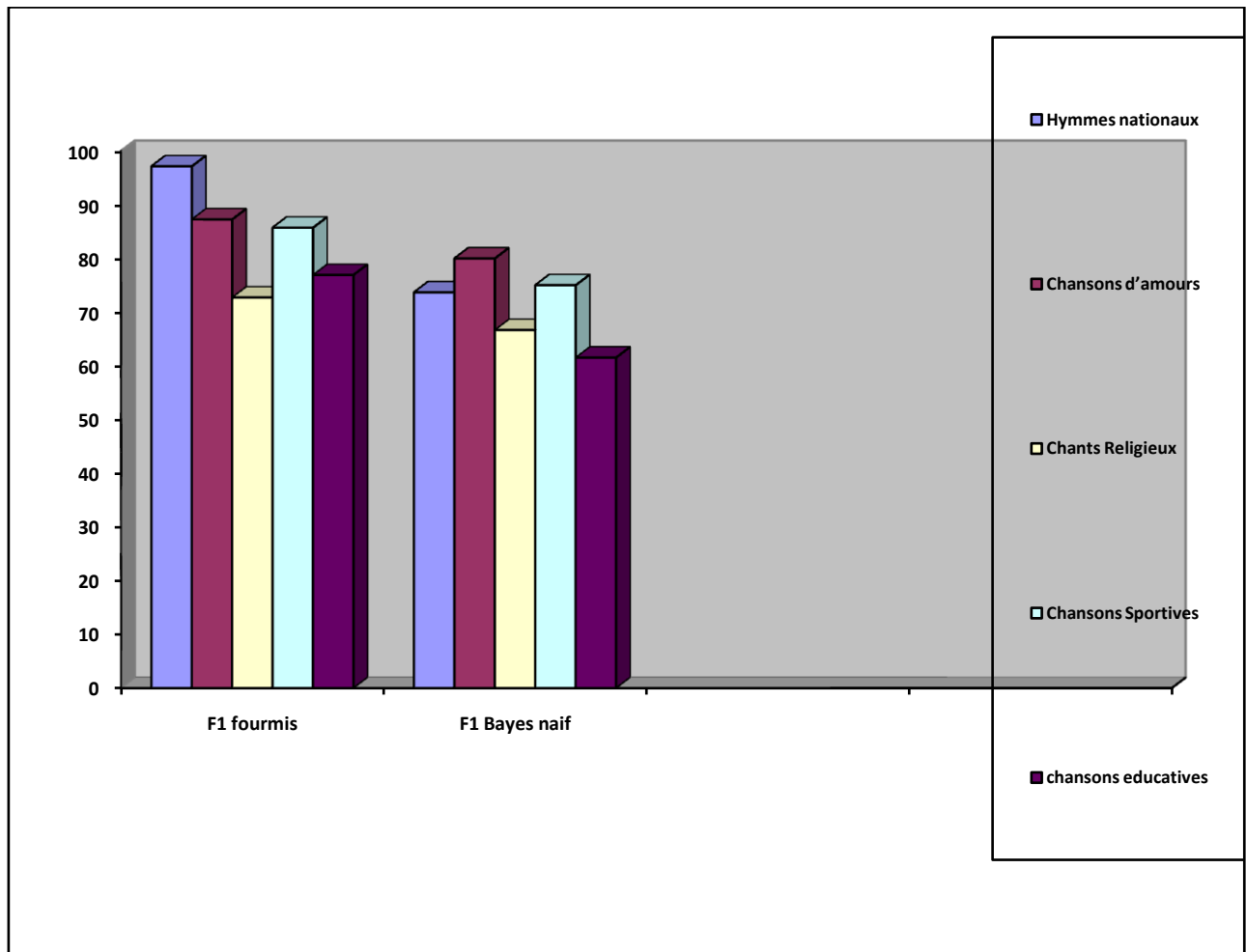


Figure 8 : Histogramme montrant la F-mesure pour chaque catégorie pour les 2 algorithmes

L'histogramme montre que l'algorithme de colonie de fourmis donne de bons résultats avec un F1 mesure nettement supérieur à celle de Bayes naïf.

III.8.2. Le temps d'exécution

Pour évaluer les performances de notre suggestion, nous faisons quelques expériences en utilisant un nombre de chansons différents à chaque exécution et nous calculons le temps d'exécution pour l'algorithme de colonie de fourmis et l'algorithme Bayes naïf.

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

Tableau 6: Relation entre le temps d'exécution et le nombre de chansons

Nombre de chansons	Temps d'exécution (ms)	
	ACO	Bayes Naïf
5	0.02	0.01
10	0.04	0.04
50	10.7	30.9
100	15.8	64
500	20.6	80.3
1000	30.5	124.1

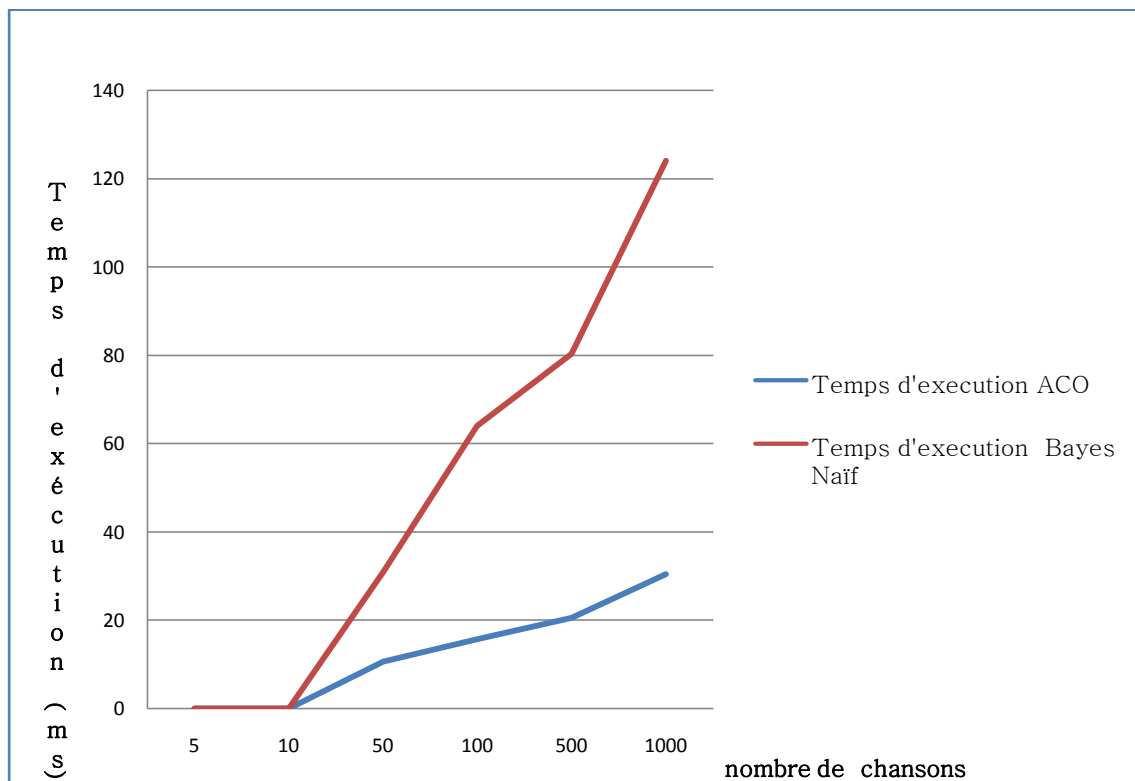


Figure 9 : Relation entre le temps d'exécution et le nombre de chansons

Partie III Application d'un algorithme de colonies de fourmis à la catégorisation de chansons

La figure 9 présente le temps d'exécution pour la classification de chansons. Les expériences montrent que pour un nombre petit de chansons l'algorithme Bayes naïf est plus rapide que l'algorithme de colonie de fourmis. Cependant, il faut noter que, pour un nombre grand de chansons l'algorithme Bayes naïf est beaucoup plus lent que l'algorithme de colonie de fourmis. Ce n'est pas une surprise puisque l'utilisation d'un modèle parallèle (colonies de fourmis) réduit le temps d'exécution et améliore la qualité de la catégorisation

III.9. Conclusion

L'utilisation de colonie de fourmis pour la classification supervisée de chansons donne de bons résultats et permet d'économiser le temps. Toutefois, reste un problème de représentation de chansons à l'aide de moyens qui favorisent la réutilisation. En plus le traitement automatique de l'indexation utilisant les colonies de fourmis ne fonctionne que sur des principes statistiques, il ne permet pas de prendre en compte les problèmes d'homonymie, de synonymie et de contextualisation liés à l'emploi du langage naturel par les utilisateurs lors de la recherche de chansons. Dans cette perspective, nous avons proposé d'explorer la méthode d'indexation conceptuelle en utilisant une ontologie de domaine de chansons qui sera discuté dans le chapitre 3.

**Chapitre III: L'indexation et la recherche
conceptuelle de chansons**

Chapitre III

Ce chapitre contient trois parties:

La première partie présente le web sémantique, la notion d'ontologie et des outils pour la création d'ontologies. Elle détaille la conception et la construction de l'ontologie de domaine pour la représentation des chansons. En fin, elle utilise l'outil Protégé pour la création et l'édition de l'ontologie.

La seconde partie expose l'indexation et la catégorisation conceptuelle et multi-langues de chansons en utilisant une ontologie de domaine, les relations et les liens sémantiques. L'approche est évaluée sur les chansons algériennes et comparée à la catégorisation en utilisant l'algorithme Bayes naïf.

La troisième partie traite la recherche conceptuelle de chansons et expose un algorithme de raffinement de requête. Une application web est développée. Des expériences sont décrites et des résultats et des discussions sont exposés à la fin.

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

I. Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

I.1. Web sémantique

I.1.1. Définition

Le Web sémantique (plus techniquement appelé *le Web de données* désigne un ensemble de technologies visant à rendre le contenu des ressources du World Wide Web accessible et utilisable par les programmes et agents logiciels, grâce à un système de métadonnées formelles, utilisant notamment la famille de langages développés par W3C.

Le web sémantique permet aux machines de comprendre la sémantique, la signification de l'information sur le web. Il étend le réseau des hyperliens entre page Web classiques par un réseau de liens entre données structurées permettant ainsi aux agents automatisés d'accéder plus intelligemment aux différentes sources de données contenus sur le Web et de cette manière d'effectuer des tâches (recherche, apprentissage, etc).

L'expression a été inventée par Tim Berners-Lee, l'inventeur du World Wide Web et directeur du World Wide Web Consortium (W3C), qui supervise le développement des technologies communes du Web sémantique. Il définit le Web sémantique comme *un web de données qui peuvent être traitées directement et indirectement par des machines pour aider leurs utilisateurs à créer de nouvelles connaissances* [84]. Il présente le Web sémantique comme une sorte d'extension du Web des documents, qui constitue une base de données à l'échelle mondiale, afin que toutes les machines puissent mieux lier les données du Web [85].

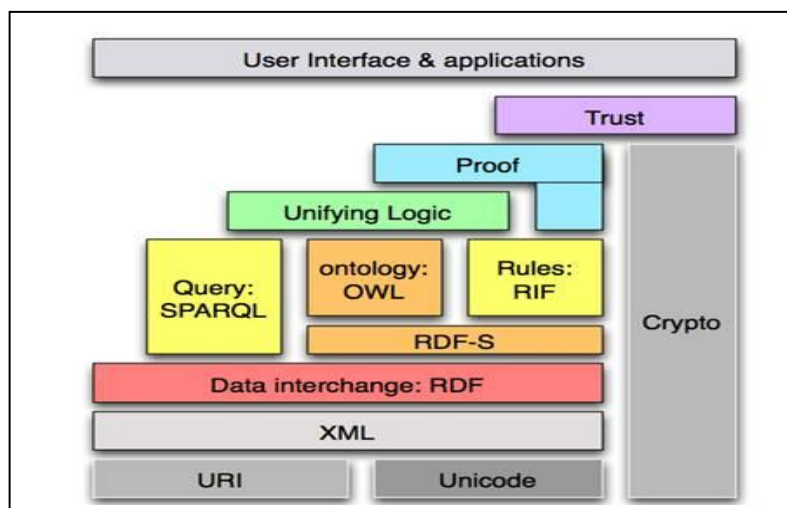


Figure 10: Les couches du standard du Web sémantique d'après [85]

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

I.1.2. Principe général

Un des principaux objectifs du web sémantique est de permettre aux utilisateurs d'utiliser la totalité du potentiel du Web: ainsi, ils pourront trouver, partager et combiner des informations plus facilement. Aujourd'hui tout le monde est capable d'utiliser les forums, d'utiliser des réseaux sociaux, de chatter, de faire des recherches ou même d'acheter différents produits. Néanmoins, il serait mieux que la machine fasse tout ceci à la place de l'homme, car actuellement, les machines ont besoin de l'homme pour effectuer ces tâches. La raison principale est que les pages Web actuelles sont conçues pour être lisibles par des êtres humains et non par des machines. Le Web sémantique a donc comme principal objectif que ces mêmes machines puissent réaliser seules toutes les tâches fastidieuses comme la recherche ou l'association d'informations et d'agir sur le Web lui-même [85].

Les documents traités par le Web sémantique contiennent non pas des textes en langages naturel (français, espagnol, chinois, etc.) mais des informations formalisées pour être traitées automatiquement. Ces documents sont générés, traités, échangés par des logiciels. Ces logiciels permettent souvent, sans connaissance informatique de:

- ✓ Générer des données sémantiques à partir de la saisie d'information par les utilisateurs;
- ✓ Agréger des données sémantiques afin d'être publiées ou traitées;
- ✓ Publier des données sémantiques avec une mise en forme personnalisée ou spécialisée;
- ✓ Echanger automatiquement des données sémantiques automatiquement, sans saisie humaine, à partir de règles d'inférence [84].

I.2. Les ontologies pour le web sémantique

I.2.1. Définition d'une ontologie

La première définition pour les ontologies dans le domaine de l'intelligence artificielle est donnée par T. Gruber dans [86]: *une ontologie est une spécification explicite d'une conceptualisation.*

- Le terme conceptualisation: fait référence à un modèle abstrait d'une partie de monde réel permettant d'identifier les concepts pertinents de ce monde.
- Le terme explicite: signifie que l'identification de la structure des concepts ainsi que les contraintes sur leur utilisation sont définies d'une manière claire et précise.

I.2.2. Les ontologies et la représentation des connaissances

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

Utilisées pour le besoin de représentation de connaissances dans les systèmes informatiques, les ontologies sont définies en se basant sur la proposition de T. Gruber [86].

La construction d'une ontologie n'intervient donc qu'après une étape de conceptualisation qui consiste à identifier, au sein d'un corpus, les connaissances spécifiques au domaine de connaissances à représenter. « A conceptualization is an abstract, simplified view of the world that we wish to represent for some purpose » [87].

N.Guarino affine la définition de T.Gruber en considérant les ontologies comme des spécifications partielles et formelles d'une conceptualisation [88]. Les ontologies sont formelles car exprimés sous forme logique, et partielles car une conceptualisation ne peut pas toujours être entièrement formalisée dans un cadre logique à cause des limites des langages logiques à exprimer certaines sémantiques conçues dans l'ontologie.

Il est donc nécessaire de pouvoir construire une première modélisation semi-formelle, partiellement cohérente, correspondant à une conceptualisation semi-formalisée. On parle alors d'*ontologie conceptuelle*, semi-formelle, et le processus de spécification en question est appelé *ontologisation* [89]. Dans tous les cas, il est nécessaire de traduire cette ontologie dans un langage formel et opérationnel de représentation des connaissances afin de pouvoir l'utiliser dans une machine. Le langage cible doit donc permettre de représenter les différents types de connaissances (connaissances terminologiques, fait, règles et contraintes) et de manipuler ces connaissances à travers des mécanismes adaptés à l'objectif opérationnel du système conçu. Ce processus de traduction est appelé *opérationnalisation*. Une base de connaissance contient les connaissances utilisées dans un système à base de connaissances. Ces connaissances sont formalisées et des mécanismes permettent de gérer la base pour consulter des connaissances ou en ajouter. L'ontologie est un système de termes primitifs utilisés dans la construction de systèmes artificiels ou une *métabase de connaissance* [90]. L'ontologie est un type particulier de modèle de connaissances [91] permettant de construire une ou plusieurs bases de connaissances du même type.

I.2.3. Les constituants d'une ontologie

Les connaissances portent sur des objets auxquels on se réfère à travers des concepts. Un concept peut représenter un objet matériel, une notion, une idée [92]. Un concept peut être divisé en trois parties: un terme (ou plusieurs), une notion et un ensemble d'objets.

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

La notion, également appelée intension du concept, contient la sémantique du concept, exprimée en termes de propriétés et d'attributs, de règles et de contraintes.

L'ensemble d'objets, également appelé extension du concept, regroupe les objets manipulés à travers le concept; ces objets sont appelés instances du concept.

Il est à noter qu'un concept peut très bien avoir une extension vide. Il s'agit alors d'un concept générique.

Le langage naturel contient de nombreux termes désignant plusieurs concepts sémantiques différents (par exemple « table » pour un meuble et « table » pour un tableau de valeurs numériques), et de telles ambiguïtés ne sont pas gérables en machine, où on identifie généralement un concept à l'aide de ses termes.

Néanmoins, la restriction à un domaine de connaissances permet généralement d'éviter les homonymies de concepts. Il apparaît par contre souhaitable de gérer les synonymies et de permettre la désignation d'un concept par plusieurs termes, pour assurer une plus grande souplesse d'utilisation de l'ontologie [93].

Un autre débat non tranché porte sur les propriétés contenues dans les intensions des concepts. En effet, certaines propriétés sont essentielles à la caractérisation d'un concept, dans le sens où la suppression de cette propriété entraîne la disparition du concept en tant que tel. D'autres propriétés peuvent cependant être considérées pour caractériser le concept dans un contexte donné. Ces propriétés ne sont vraies que dans ce cadre et leur disparition ne modifie pas le concept.

Les concepts manipulés dans un domaine de connaissances sont organisés au sein d'un réseau de concepts. L'ensemble des concepts y est structuré hiérarchiquement et les concepts sont liés par des propriétés conceptuelles.

La propriété utilisée pour structurer la hiérarchie des concepts est la *subsumption* (*super-concept*), qui lie deux concepts:

Un concept C_1 subsume un concept C_2 si toute propriété sémantique de C_1 est aussi une propriété sémantique de C_2 , c'est-à-dire que C_2 est plus spécifique que C_1 .

L'extension d'un concept est forcément plus réduite que celle d'un concept qui le subsume. Son intension est par contre plus riche.

Une liste des principales propriétés pouvant être associées à un concept est donnée ci-dessous. Les propriétés portant sur les extensions de deux concepts sont utiles dans le cas où on ne définit un concept que par son intension, mais où l'on veut préciser qu'aucune instance commune aux deux concepts ne doit être créée ultérieurement sans respecter certaines propriétés.

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

Les propriétés portant sur deux concepts sont:

- **L'équivalence:** deux concepts sont équivalents s'ils ont même extension.
- **La disjonction:** (on parle aussi d'incompatibilité) deux concepts sont disjoints si leurs extensions sont disjointes. Par exemple: « homme » et « femme »;
- **La dépendance:** un concept C_1 est dépendant d'un concept C_2 si pour toute instance de C_1 il existe une instance de C_2 qui ne soit ni partie ni constituant de l'instance de C_2 . Par exemple: « parent » est un concept dépendant de « enfant » (et vice-versa).

Ces propriétés ont été formalisées, afin d'être traduites dans des langages d'ontologie.

En plus des propriétés, l'intention d'un concept peut contenir des attributs. Un attribut peut être une instance de concept.

Si certains liens conceptuels existant entre les concepts peuvent s'exprimer à l'aide de propriétés portées par les concepts, d'autres doivent être représentés à l'aide de relations autonomes. Une relation permet de lier des instances de concepts, ou des concepts génériques. Elles sont caractérisées par un terme (voire plusieurs) et une signature qui précise le nombre d'instances de concepts que la relation lie, leurs types et l'ordre des concepts, c'est-à-dire la façon dont la relation doit être lue. Par exemple, la relation «écrit» lie une instance du concept «personne» et une instance du concept «texte», dans cet ordre.

Tout comme les concepts, les relations peuvent être spécifiées par des propriétés ou dans une liste, non exhaustive. Les relations sont organisées de manière hiérarchisée à l'aide de la propriété de subsomption. Les propriétés fondamentales d'une relation sont:

- **Les propriétés algébriques:** symétrie, réflexivité, transitivité.
- **La cardinalité:** nombre possible de relations de même type entre les mêmes concepts (ou instances de concept). Les relations portant une cardinalité représentent souvent des attributs. Par exemple, une «pièce» a au moins une «porte», un «humain» a entre zéro et deux «jambes».

Les propriétés liant deux relations sont:

- **L'incompatibilité:** deux relations sont incompatibles si elles ne peuvent lier les mêmes instances de concepts. Par exemple, les relations «être rouge» et «être vert» sont incompatibles;
- **L'inverse:** deux relations binaires sont inverses l'une de l'autre si, quand l'une lie deux instances I_1 et I_2 , l'autre lie I_2 et I_1 . Par exemple, les relations «a pour père» et «a pour enfant» sont inverses l'une de l'autre.

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

- **L'exclusivité:** deux relations sont exclusives si, quand l'une lie des instances de concepts, l'autre ne lie pas ces instances, et vice-versa. L'exclusivité entraîne l'incompatibilité. Par exemple, l'appartenance et le non appartenance sont exclusifs.

Les propriétés liant une relation et des concepts sont:

- **Le lien relationnel:** (propriété proposée par Kassel [74]). Il existe un lien relationnel entre une relation R et deux concepts C_1 et C_2 si, pour tout couple d'instances des concepts C_1 et C_2 , il existe une relation de type R qui lie les deux instances de C_1 et C_2 .
- **La restriction de relation:** (propriété proposée par Kassel [74]). Pour tout concept de type C_1 , et toute relation de type R liant C_1 , les autres concepts liés par la relation sont d'un type imposé. Par exemple, si la relation «mange» portant sur une «personne» et un «aliment» lie une instance de «végétarien», concept subsumé par «personne», l'instance de «aliment» est forcément instance de «végétaux».

Ces propriétés associées aux concepts et relations complètent la sémantique différentielle de l'ontologie au sens où elles contribuent à préciser les liens et différences entre les primitives cognitives du domaine de connaissances. Les aspects différentiels et référentiels sont cependant fortement imbriqués et la construction d'une ontologie est un processus complexe qui demande en particulier des choix de représentation délicats.

I.2.4. Les langages d'ontologies

- **KIF:** KIF est un langage basé sur les prédicats du premier ordre avec des extensions pour représenter des définitions et des méta-connaissances, la logique du premier ordre étant un langage de bas niveau pour l'expression d'ontologies. Une extension du langage KIF, ONTOLINGUA, est utilisée dans le serveur d'édition d'ontologies, Ontolingual du même nom.
- **RDF et RDF Schéma:** RDF est un modèle de données pour représenter les objets et les relations entre eux, fournissant une sémantique simple pour ce modèle qui peut être représenté dans une syntaxe XML (Extended Markup Language). RDF-Schéma est un langage de définition de vocabulaire pour la description de propriétés et de classes représentées par des ressources RDF. RDF-S permet de définir des graphes de triplets RDF, avec une sémantique de généralisation/hiérarchisation de ces propriétés et de ces classes

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

- **DAML + OIL:** Dans l'optique d'une utilisation d'ontologies sur le Web, le langage RDF-S a été enrichi par l'apport du langage OIL (Ontology Interchange Language) qui permet d'exprimer une sémantique à travers le modèle des frames tout en utilisant la syntaxe de RDF-S. OIL offre de nouvelles primitives permettant de définir des classes à l'aide de mécanismes ensemblistes issus des logiques de description (intersection de classes, union de classes, complémentaire d'une classe). Il permet également d'affiner les propriétés de RDF-S en contraignant la cardinalité ou en restreignant la portée [95].

Le langage OIL a été fusionné avec le langage DAML pour former le DAML+OIL. DAML est conçu pour permettre l'expression d'ontologies dans une extension du langage RDF. Il offre les primitives usuelles d'une représentation à base de frames et utilise la syntaxe RDF [96]. L'intégration de OIL rend possible les inférences compatibles avec les logiques de description, essentiellement les calculs de liens de subsomption.

- **OWL:** La combinaison de RDF/RDF-S et de DAML+OIL a permis l'émergence d'OWL (Web Ontology Language), un langage standard de représentation de connaissances pour le Web. Développé par le groupe de travail sur le Web Sémantique du W3C, OWL peut être utilisé pour représenter explicitement les sens des termes des vocabulaires et les relations entre ces termes. OWL vise également à rendre les ressources sur le Web aisément accessibles aux processus automatisés [97], d'une part en les structurant d'une façon compréhensible et standardisée, et d'autre part en leur ajoutant des méta-informations.

Pour cela, OWL a des moyens plus puissants pour exprimer la signification et la sémantique que XML, RDF, et RDF-S. De plus, OWL tient compte de l'aspect diffus des sources de connaissances et permet à l'information d'être recueillie à partir de sources distribuées, notamment en permettant la mise en relation des ontologies et l'importation des informations provenant explicitement d'autres ontologies.

I.2.5. Les outils de construction d'ontologies

De nombreux outils de construction d'ontologies utilisent des formalismes variés et offrent différentes fonctionnalités. Seuls les plus connus seront cités ici. Tous ces outils offrent des supports pour le processus de création d'ontologies, mais peu offrent une aide à la conceptualisation: **KAON (Karlsruhe Ontology and Semantic Web)**, **ODE**, **ONTOLINGUA**,

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

OILED, WEBONTO et **Protégé**. Par la suite, on détaille Protégé qui sera utilisé pour créer notre ontologie de domaine de chansons.

Protégé est un système auteur pour la création d'ontologies. Il a été créé à l'université Stanford et est très populaire dans le domaine du Web sémantique et au niveau de la recherche en informatique. Protégé est développé en Java. Il est gratuit et à code source libre. Protégé peut lire et sauvegarder des ontologies dans la plupart des formats d'ontologies: RDF, RDFS, OWL, etc. Plusieurs systèmes auteurs lui livrent une concurrence telle que Hozo, Onto Edit et Swoop. Plusieurs utilisateurs préfèrent Protégé pour sa convivialité.

I.3. Construction de l'ontologie de chansons

Pour représenter la complexité des chansons en vue de pouvoir effectuer une recherche d'information au milieu des entrepôts de chansons, nous sommes tenus de représenter les chansons dans un formalisme qui puisse restituer les métadonnées. En plus, pour faire face aux problèmes de représentations approximatives, pauvres et partielles du contenu sémantique des chansons en utilisant des techniques d'indexation traditionnelles vues dans le chapitre 2 et afin de pouvoir indexer des collections de chansons qui soient à l'échelle de la production musicale nous proposons:

- d'exploiter la spécificité de l'information musicale qui peut revêtir de multiples aspects: elle peut être constituée d'un ou plusieurs des paramètres du langage musical, ou bien être textuelle. J. Stephen Downie [17] attribue sept facettes à l'information musicale: tonale, temporelle, harmonique, timbrale, éditoriale, textuelle et bibliographique.
- Notre étude porte sur la facette textuelle (parole de chansons) et la facette bibliographique (titre, chanteur, genre..).
- d'exploiter la structure spécifique des chansons dans laquelle chaque partie (mélodie, parole) joue un rôle différent. Les paroles (facette textuelle) sont divisées en plusieurs parties: titre, introduction, verset, refrain et pont. Parmi eux, les plus importantes parties sont le titre et le refrain qui est un verset répété au moins deux fois sans ou avec une petite différence entre les répétitions et contient pratiquement le sens et l'émotion d'une chanson. C'est la partie que les auditeurs ont tendance à se rappeler.
- d'utiliser les caractéristiques de l'artiste (facette bibliographique) puisque chaque artiste a tendance à composer ou à chanter un genre particulier de musiques.

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

- d'introduire des notions et des technologies du web sémantique pour créer une ontologie de domaine contenant plusieurs concepts, attributs et relations pour représenter et décrire nos données et stocker nos chansons, puis nous entamons l'indexation des chansons en utilisant les outils textuels et l'ontologie de domaine, après nous créons un navigateur d'ontologie OWL permettant la visualisation graphique de notre ontologie et nous présentons une nouvelle approche de recherche et d'indexation conceptuelle pour indexer et rechercher une chanson .
- de développer et d'utiliser un algorithme de raffinement de requêtes qui transforme une requête en langage naturel en SPARQL puis effectuer la recherche.

Pour créer l'ontologie de domaine de chansons, nous nous sommes inspirés du travail présenté dans [47] décrivant une ontologie de musique et fournissant les principaux concepts et propriétés pour décrire la musique (artistes, albums, chansons ...) sur le Web sémantique. En outre, dans notre ontologie, nous considérons les paroles.

Le concept de la chanson et ses propriétés permet le stockage de toutes les chansons tandis que les autres concepts relèvent du domaine. Chaque chanson appartient à un thème qui est indexé par ses mots-clés.

Les concepts de notre ontologies sont: chanson, album, enregistrement, époque, genre, instrument, Mot, Refrain, Parole, Mots-clés, Mélodie, Performance, Personne, Chanteur, Compositeur, Ecrivain, Région et Thème et chaque concept peut avoir des sous concepts et il a des attributs et peut avoir des attributs hérités d'un concept père.

I.3.1. Liste des concepts de l'ontologie de domaine

Dans le tableau qui suit on présente les concepts de l'ontologie de domaine ainsi que leurs définitions:

Tableau 7: Liste des concepts

Concept	Sur concept	Définition du concept
Music	Ontologie	Music est un concept d'application, Elle représente l'ontologie de domaine
Chanson	Music	Une Chanson est un concept music, Elle a un, titre, un genre, une époque, un thème, une langue et des paroles. Elle est chantée par un seul chanteur ou un groupe de chanteurs.
Album	Music	Un Album est un concept music, il peut être un disque vinyle, un CD ou un ensemble de fichiers numériques

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

		présenté ensemble et regroupant un ensemble de titres d'un chanteur, d'un groupe ou d'un musicien. Les albums consacrés à un genre musical et contenant plusieurs titres de différents interprètes et les albums rassemblant plusieurs morceaux d'un même artiste enregistrés à l'origine sur différents disques sont appelés compilations.
Enregistrement	Music	Un Enregistrement est un concept music, il peut être live, remix ou compilation.
Epoque	Music	Une Epoque est un concept music, elle peut être 2000, 60, 70, 80, 90, Actuelle, Révolution, Avant Révolution et elle représente l'époque à laquelle une chanson est chantée pour la première fois.
Genre	Music	Genre est un concept music, il peut être pour la chanson Algérienne: AraboAndalouse, Chaoui, Diwane, Genawa, Kabyle, Moderne, PopulaireCitadine, Raï, Staïfi. Folklorique.
Instrument	Music	Instrument est un concept music qui peut être Cordes, Percussions, Vents, Guitare_classique, Guitare_electrique, piano, synthétiseur, Guitare_folk.
Mot	Music	Mot est un concept music qui peut être Mot-Refrain, Mot-Parole, Refrain, Titre.
Mots-clés	Mot	Mots-clés est un ensemble de mots qui exprime la thématique de la chanson
Refrain	Mot	Refrain est un ensemble de mots qui désigne les mots figurant dans les refrains avec leurs fréquences.
Parole	Mot	Parole est un ensemble de mots qui se trouve dans les paroles.
Mélodie	Music	Mélodie est un concept music qui est Composition musicale faite sur le texte d'un poème.
Performance	Music	Performance est un concept music.
Personne	Music	Personne est un concept music qui peut être Compositeur, Ecrivain ou un Chanteur.
Compositeur	Personne	Compositeur est une personne qui compose la musique de la chanson.
Chanteur	Personne	Chanteur est une personne qui chante la chanson.
Ecrivain	Personne	Ecrivain est une personne qui écrit les paroles de la chanson.
Région	Music	Région est un concept music qui peut être East, Nord, Sud, Ouest, Etranger.
Thème	Music	Thème est un concept music qui peut être Fête, National, Religieux, Sentimental, Social, Sportif, Historique, Nature.

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

I.3.2. Liste des attributs

Dans le tableau qui suit on va présenter pour chaque concept ses attributs, et s'il a des attributs hérités d'un concept père.

Tableau 8: Liste des attributs

Concept	Hérite les attributs de concept	Attributs	Commentaires
Chanson	Music	Durée chanson	Le temps qu'a duré la chanson
		Titre	Le titre de la chanson
		Langue	La langue des paroles de la chanson
Album	Music	Titre	Titre de l'album
		Date sortie	Date de sortie de l'album
		Nbre chansons	Le nombre de chansons dans l'album
		Nbre exemplaire	Le nombre d'exemplaire de l'album
Epoque	Music	Durée	Par durée on désigne la durée totale de l'album
		Date début	La date de début de la période
Enregistrement	Music	Date fin	La date de fin de la période
		Type_Enregistrement	Un Type d'Enregistrement peut être live, remix ou compilation
Genre	Music	Type_Genre	Type de chanson qui peut être AraboAndalouse, Chaoui, Diwane, Kabyle Folklorique, Genawa, Moderne, PoulareCitadine, Raï, Staïfi
Instrument	Music	Type_Instrument	Type de l'instrument qui peut être Cordes, Percussions, Vents.
		Désignation	La désignation de l'instrument.
Mot	Music	Définition	La définition d'un mot
		Lemme	Un lemme est une unité autonome constituante du lexique d'une langue
		Mot-forme	Un mot-forme est un mot, n'importe lequel, considéré dans son aspect non sémantique mais graphique
Mots-clés	Mot	Thématique	Une thématique exprime le thème de la chanson

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

Refrain	Mot	Nbre répétition-mot	Le nombre de répétitions du mot constituant les refrains
Parole	Mot	Langue	La langue dans laquelle les paroles sont écrites
		Mots-clés	Ensemble de mots qui ont une importance particulière à partir desquelles on peut définir la thématique de la chanson
Personne	Music	Nom	Le nom de la personne
		Prénom	Prénom de la personne
		Surnom	Le surnom de la personne
		Date naissance	La date de naissance de la personne
		Date décès	Date de décès de la personne s'il est décédé
		Nationalité	La nationalité de la personne
Compositeur	Personne	Site	Le site personnel du compositeur
Chanteur	Personne	Site	Le site personnel du chanteur
Ecrivain	Personne	Site	Le site personnel de l'écrivain

I.3.3. Liste des relations

Dans le tableau qui suit on va décrire les relations qui existent entre les différents concepts de l'ontologie.

Tableau 9 : Liste des relations

Relation	Prédécesseur	Successeur	Définition
Pendant	Epoque	Chanson	Une chanson est sortie pendant une époque si et seulement si elle est chantée pour la première fois durant cette époque
Style	Genre	Chanson	Une chanson a un et un seul style qui est son genre
Chante_genre	Genre	Chanteur	Un genre est chanté par un chanteur
Has_genre	Chanteur	Genre	Un chanteur chante un genre de musique
Appartient	Région	Chanson	Une chanson appartient à une région
Élément	Album	Chanson	Au moins une chanson est élément d'un album
Enregistré_sur	Album	Enregistrement	Au moins un album est enregistré sur un enregistrement
Performance	Mélodie	Parole,	C'est une association ternaire entre

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

mélodie		Chanson	les classes (mélodie, parole, chanson), la multiplicité indique le nombre d'objets mélodie qui peuvent apparaître dans l'association avec une paire particulière d'objets parole, chanson.
Performance chanson	Chanson	Mélodie, Parole	C'est une association ternaire entre les classes (chanson, parole, mélodie), la multiplicité indique le nombre d'objets chanson qui peuvent apparaître dans l'association avec une paire particulière d'objets parole, mélodie.
Performance Parole	Parole	Chanson, mélodie	C'est une association ternaire entre les classes (parole, mélodie, chanson), la multiplicité indique le nombre d'objets parole qui peuvent apparaître dans l'association avec une paire particulière d'objets mélodie, chanson.
Has_thème	Thème	Chanson	Une chanson a un et un seul thème si et seulement si elle tourne autour de ce thème
Mots-clés	Thème	Mots-clés	Un ou plusieurs mots clés sont des mots clés d'un thème s'ils permettent de le décrire et l'identifier.
Chante_thème	Thème	Chanteur	Un ou plusieurs chanteurs chantent un thème.
Is a personne	Compositeur	Personne	Une personne est un compositeur si et seulement si, c'est elle qui a composé la musique.
appartient	Refrain	Chanson	Refrain appartient à une ou plusieurs chansons
Contient	Refrain	Mot	Refrain contient un ou plusieurs mots
Has_refrain	Chanson	Refrain	Une ou plusieurs chansons has_refrain refrain
Composé de	Mot	Mots-clés	Mots-clés est composé d'un ou plusieurs mots
Décrit	Mots-clés	Parole	Parole est décrite par des mots-clés
Constitué	Parole	Mot	Un ou plusieurs mots constituent parole
Is a personne	Chanteur	Personne	Une personne est un chanteur si et

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

			seulement si c'est elle qui chante
Chante	Chanteur	Chanson	Un chanteur chante au moins une chanson
Is a personne	Ecrivain	Personne	Une personne est un écrivain si et seulement si c'est elle qui écrit les paroles de la chanson
Ecrit	Ecrivain	Parole	Parole est écrite par un et un seule écrivain
Synonyme	Mot	Mot	Un mot à zéro ou plusieurs mots synonymes
Homonyme	Mot	Mot	Un mot à zéro ou plusieurs mots homonymes
Compose	Compositeur	Mélodie	Un ou plusieurs compositeurs composent une ou plusieurs mélodies
Produit	Instrument	Mélodie	Un ou plusieurs instruments produisent une mélodie

I.3.4. Diagramme de classes de l'ontologie de domaine pour la description des chansons

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

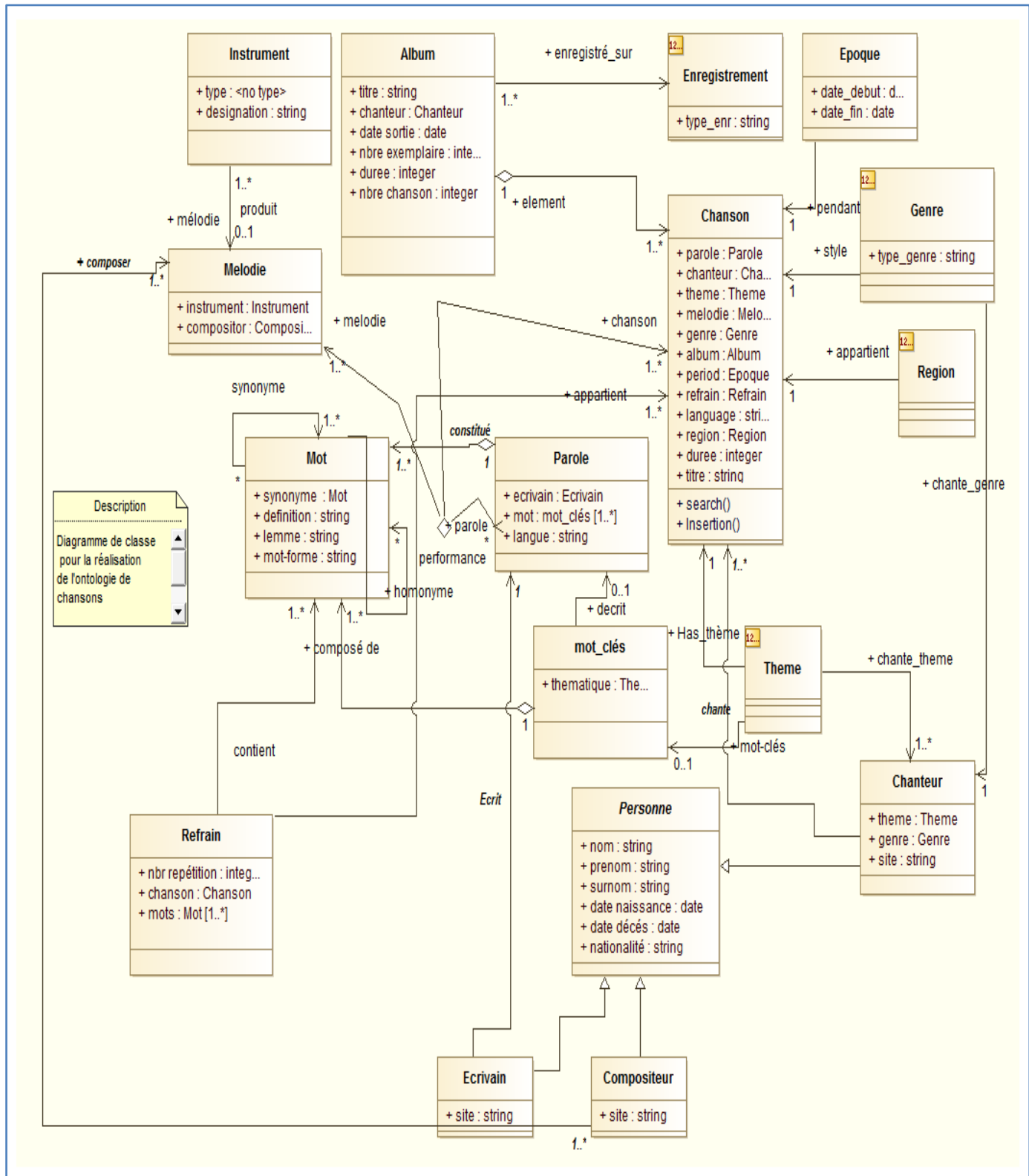


Figure 11: Diagramme de classes de l'ontologie de domaine

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

I.4. Création de l'ontologie de domaine de chansons

Pour créer notre ontologie nous avons utilisé Protégé. Protégé est un système auteur pour la création d'ontologies. Il a été créé à l'université Stanford et il est très populaire dans le domaine du Web sémantique et au niveau de la recherche en informatique. Protégé est développé en Java. Il est gratuit et peut lire et sauvegarder des ontologies dans la plupart des formats d'ontologies: RDF, RDFS, OWL, etc.

La racine de tous les concepts du domaine est « Music ». Ce nœud possède toutes les caractéristiques de la structure de l'ontologie de domaine, l'hierarchie de classes avec ses classes et ses instances, ses relations, propriétés, contraintes, annotation concernant le domaine de connaissance. Tous les termes ou les descripteurs seront modélisés dans ce nœud.

Les lexicalisations des concepts du nœud music apparaissent comme des instances de la classe lexicalisation. Cette approche a été choisie plutôt que l'utilisation de rdfs: label pour chaque concept pour représenter ses lexicalisations dans un langage particulier. Cela nous permet de gérer la représentation des langues multiples. La modélisation de la lexicalisation sera traduite par l'initialisation des relations entre les lexicalisations différentes qui décrivent un concept.

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

I.4.1. Création des classes de l'ontologie de domaine de chansons

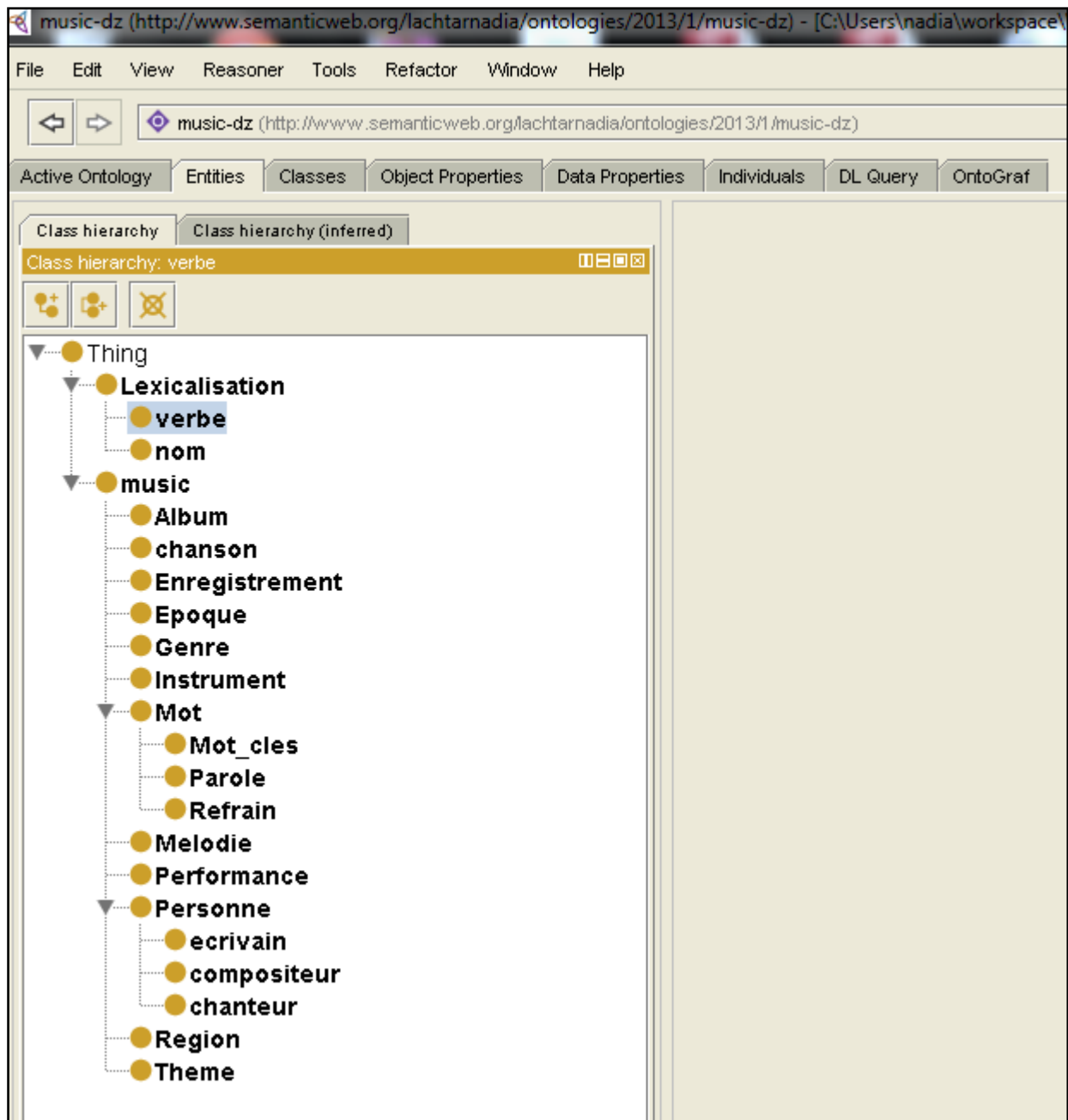


Figure 12: Création des classes de l'ontologie de chansons

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

I.4.2. Création des propriétés (les attributs) de l'ontologie de domaine de chansons

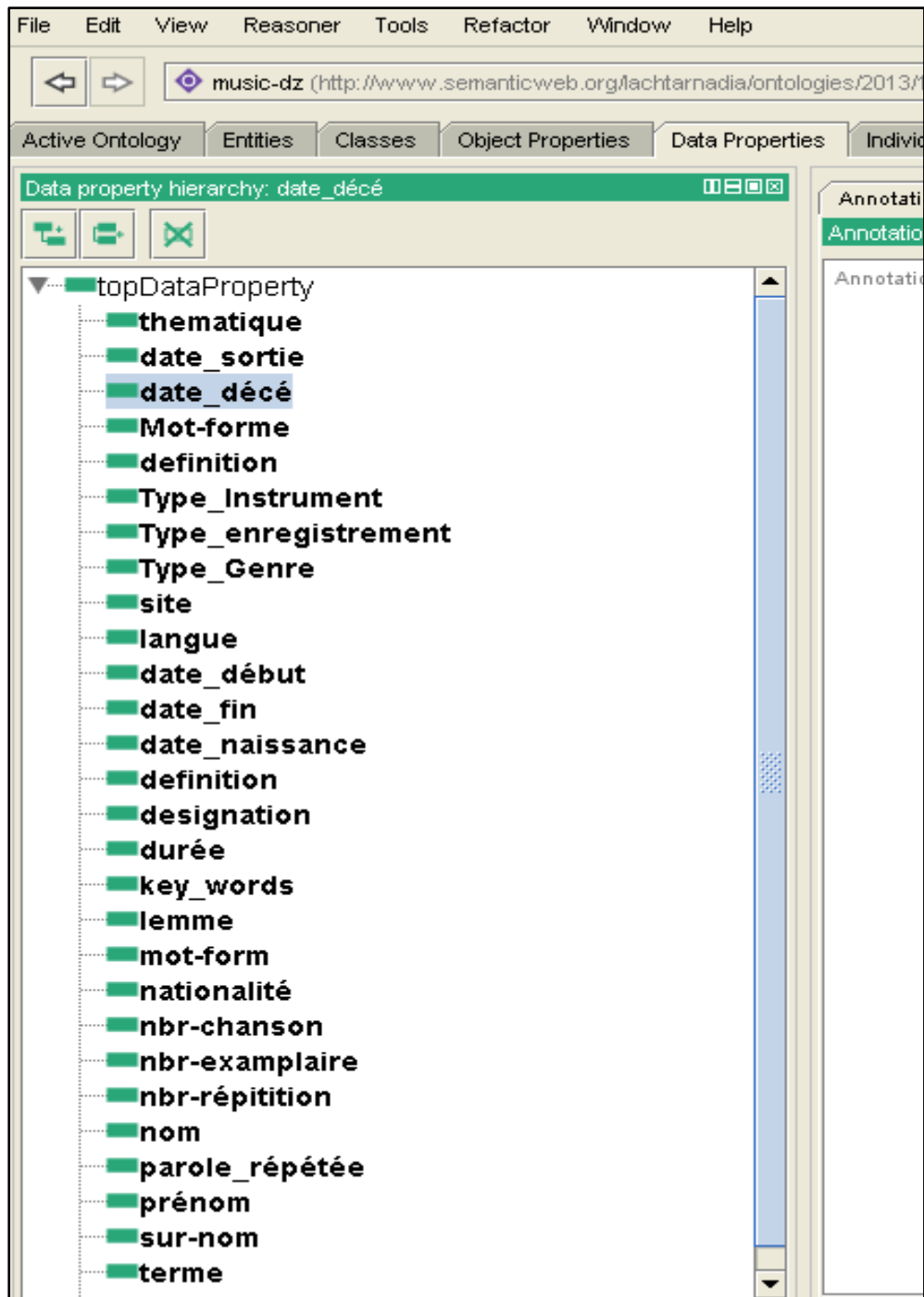


Figure 13: Création des propriétés de l'ontologie de chansons

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

I.4.3. Création des relations de l'ontologie de domaine de chansons

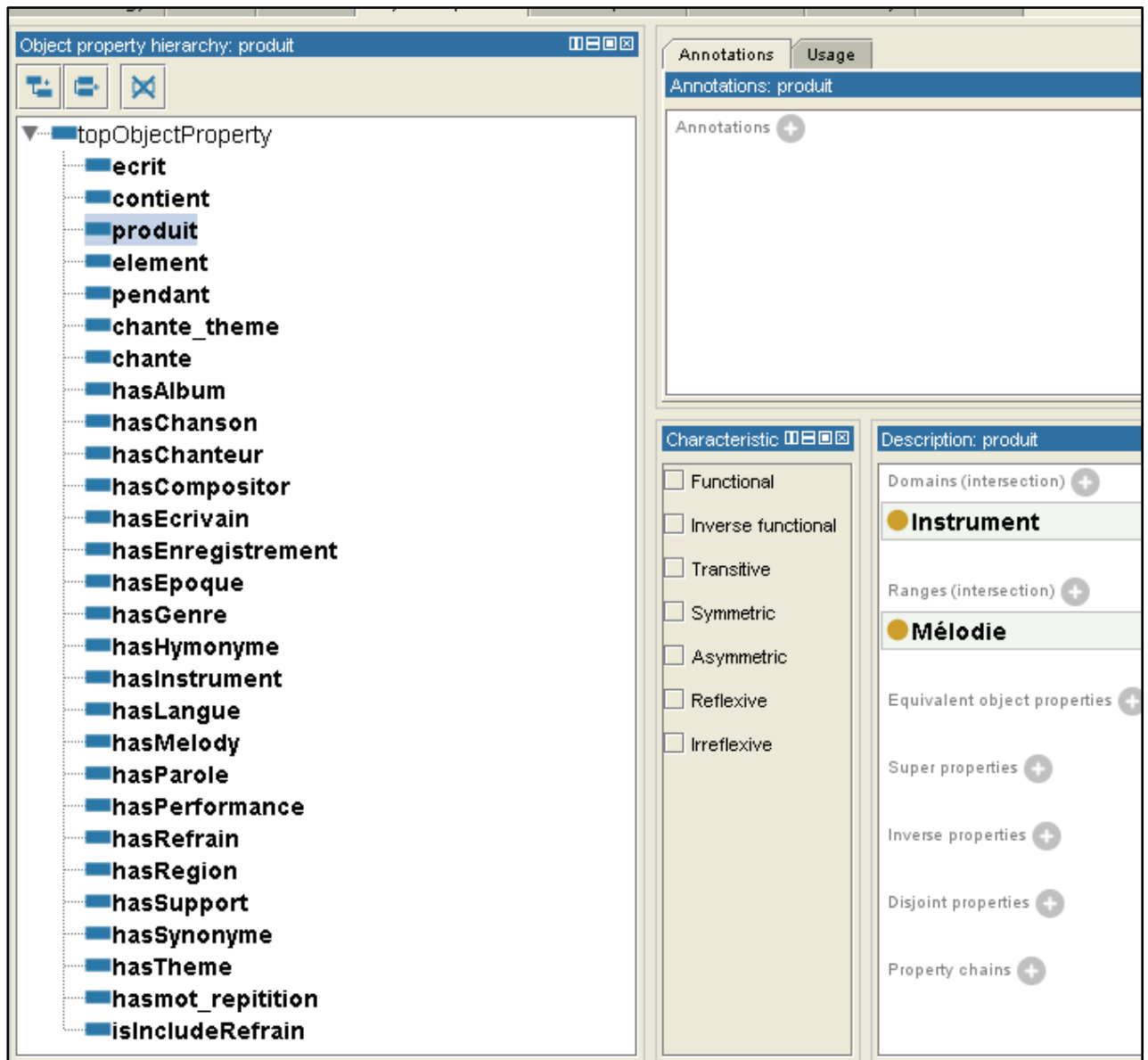


Figure 14: Création des relations de l'ontologie de chansons

Après avoir générer le code OWL correspondant à notre ontologie on a passé à son exploitation dans un programme JAVA, on a utilisé l'API Jena qui sert de lien entre un code OWL et un programme JAVA.

Jena est un ensemble d'outils (API) open source développé par HP LabsSemantic Web programme, permettant de lire et de manipuler des ontologies décrites en RDFS ou en OWL et d'y appliquer certains mécanismes d'inférences.

Partie I: Conception et création d'une ontologie de domaine de chansons en utilisant les outils du Web sémantique et des ontologies

Pour la recherche d'information musicale et l'identification de thème d'une chanson. Nous avons utilisé SparQLjena.

SparQL: abréviation de SparQL Protocol and RDF QueryLanguage signifiant Protocole SparQL et langage de requête RDF. L'utilisation de SparQL permet de standardiser l'interrogation de dépôts de données (ou graphe de données) d'où un gain de temps dans le développement de nouvelles applications.

I.5. Conclusion

Dans cette partie, nous avons présenté le principe du Web sémantique et la notion d'ontologie. Nous avons exposé les langages et les outils de construction d'ontologies. Nous avons utilisé l'outil Protégé pour créer notre ontologie de domaine de chansons dans une perspective de l'utiliser pour l'indexation et la recherche conceptuelle de chansons, l'objectif de la partie 2 et la partie 3 de ce chapitre.

Partie II La catégorisation conceptuelle multi-langues de chansons

II. La catégorisation conceptuelle multi-langues de chansons

II.1. Introduction

Cette partie aborde le problème de l'indexation des chansons et en particulier, la catégorisation d'une chanson en fonction de son thème en utilisant l'ontologie de domaine créée précédemment dans la section 4 partie 1 de ce chapitre. En effet, il est fréquent que les demandes sur le web soient liées à un thème pas une chanson.

II.2. Conception générale du système d'indexation et de catégorisation de chansons

II.2.1. Lexicalisation

Pour représenter le domaine de chansons en plusieurs langues, nous avons introduit les lexicalisations. Toutes les informations lexicales sont incluses dans le concept lexicalisation. Chaque terme (chaque lexicalisation ou mot) qui décrit un concept du domaine dans une langue spécifique est modélisé comme une instance du concept lexicalisation. Les instances sont des instances du sous concepts nom et du sous concept verbe. Les lexicalisations représentent les mots-clés définis à partir d'un corpus de chansons. Les termes sont associés à un concept à travers deux relations : la relation « has_lexicalisation » et la relation inverse « means »

Un concept du domaine sera généralement associé à plusieurs instances de lexicalisation; au moins une pour chaque langage dans laquelle il est disponible et les autres pour des synonymes contextuels et les synonymes. Pour associer deux termes (ou lexicalisations) entre eux, on introduit la relation « has_related_term ». Cette propriété d'objet est la super propriété d'objet de toutes les relations entre les termes.

Initialement, on identifie deux associations possibles entre les termes. Un terme peut avoir:

- Un ou plusieurs traductions
- Un ou plusieurs synonymes ou synonymes contextuelles par langage.

Le processus d'indexation consiste à analyser les paroles de chansons pour créer et attacher un ensemble de termes et leurs synonymes contextuels (mots-clés) pour chaque concept du domaine. Les mots-clés décrivant chaque concept seront plus facilement exploités par le système durant le processus ultérieur de l'identification de thème. Par exemple, l'un des termes liés au thème "sentimental" est la femme et parmi ses synonymes contextuels, on trouve: Echira (jeune femme dans l'ouest de l'Algérie), El Baida (la blanche), Essamra (la brune), la Blonde, erroukha (roux, de

Partie II La catégorisation conceptuelle multi-langues de chansons

roja en espagnol), echabba (la belle), l'marioula (la femme infidèle), madame. L'un des termes liés au thème "social" est l'immigration et parmi ses synonymes contextuels, on trouve: voyage, évasion, Paris, Marseille, Allemagne, aéroport, bateau, la police, douane, passeport, visa, devises, consulat, Zodiac.

II.2.2. Classification

Dans tout problème de classification, il est primordial de représenter efficacement les objets pour pouvoir bien les classer. Dans le domaine de la classification des documents, les documents sont essentiellement représentés par des vecteurs de caractéristiques. Ces caractéristiques peuvent être des mots qui existent dans le document, mais étant donné les différentes langues qui coexistent dans une chanson, nous avons choisi de représenter une chanson à classer par un vecteur de concepts et non pas de mots. Les termes sont extraits de la chanson, puis à chacun d'entre eux, nous associons le concept le plus proche dans l'ontologie de domaine. Ainsi, les descripteurs ne sont pas choisis directement dans les chansons, mais dans l'ontologie. Les chansons sont ensuite indexées par des concepts qui reflètent leur sens plutôt que des par mots souvent ambigus.

Le système de catégorisation conceptuelle se compose des éléments suivants:

- a- L'ontologie de domaine
- b- Les contextualisations

Avant d'insérer la chanson, le système doit déterminer son thème. Il procède comme suit. Si nous avons une partie de paroles d'une chanson, le système supprime tous les mots qui existent dans une stop-list et obtient une liste de mots, détermine le vecteur de caractéristiques, applique une approche de désambiguïsation si nécessaire puis classe la chanson

II.2.2.1. Vecteur de caractéristiques

Une fois la liste des mots de la chanson à classer est déterminée, le vecteur de caractéristiques est construit. Les mots sont les étiquettes des instances nom et verbe. Cette liste de mots représente les mots-clés. Les mots-clés seront appariés au concept thème ou à un de ses synonymes contextuels dans un espace conceptuel. En d'autres termes, nous devons extraire des concepts à partir des mots-clés. Chaque mot-clé sera associé à un concept de l'ontologie. Dans le cas où les mots-clés impliquent plus d'un thème, le système suit une approche de désambiguïsation.

II.2.2.2. Désambiguïsation

Partie II La catégorisation conceptuelle multi-langues de chansons

Le système demande à l'utilisateur d'insérer d'autres informations à propos de la chanson telles que: le chanteur, le genre, les instruments, la langue ou la région. Les instruments, la région et la langue sont utilisés en conjonction avec le style ou le genre d'une chanson.

Si l'utilisateur insère le chanteur de la chanson actuelle et si le chanteur existe comme une instance du concept chanteur, Le thème de la chanson sera déterminé en utilisant la relation chante_thème entre les deux instances chanteur et thème.

Donc, dans le cas d'ambiguïté entre les thèmes, des informations supplémentaires sont utilisés: par exemple, le thème le plus récurrent dans le genre Rai est sentimental, tandis que dans le genre Rap, le premier thème est social. Le thème principal d'un chanteur peut être utilisé.

Dans le cas où l'utilisateur ne connaît pas d'informations supplémentaires sur la chanson, le système procède comme suit:

Pour désambigüiser, il utilise le principe de Khan [105]. Khan, tout comme les autres chercheurs utilisant la notion de concept, démarre de l'hypothèse qu'un groupe de mots-clés qui occurrent ensemble dans un même contexte détermine des concepts appropriés pour désigner ensemble un autre concept, même si chaque mot-clé peut être individuellement ambigu. En prolongeant et formalisant l'idée du contexte afin de réaliser la désambigüisation des concepts, [105] propose un algorithme basé sur le principe de la co-occurrence.

Pour désambigüiser via une ontologie, le score d'un concept est utilisé. Ceci a pour but de sélectionner les concepts à retenir lors de la désambigüisation du sens des mots dans la chanson.

Dans l'ontologie de domaine de chansons utilisée, un concept est composé d'une liste complémentaire de synonymes $C_i = (l_1, l_2, \dots, l_i, \dots, l_n)$. Les mots-clés de la chanson annotée sont appariés avec chaque mot-clé l_j du concept. Le calcul du score pour un l_j élément (*score_element*) du concept C_i est basé sur le nombre de mots-clés reconnus:

$$score_element_{ij} = \frac{\sum_{i=1}^n \text{mots} - \text{cles de } l_j \text{ appariés}}{\sum_{i=1}^n \text{mots} - \text{cles de } l_j} \quad (1)$$

En fait cette mesure de Khan applique le principe de Lesk [106] sur le recouvrement dans les définitions des mots à comparer si l'on considère que la liste des synonymes d'un mot représente sa définition.

Le plus grand de ces scores est retenu comme score pour tout le concept:

$$Score_concept_i = \max_{j, 1 \leq j \leq n} Score_element_{ij} \quad (2)$$

Le thème de la chanson est le concept qui a le plus grand score.

Partie II La catégorisation conceptuelle multi-langues de chansons

II.2.2.3. Classification

Considérons l'extrait de parole "ndir rap (je chante rap) de Lotfi Double Kanon de l'album Dangerous".

Après élimination des mots vides (je, de, l') qui existe dans la stop-list, on obtient la liste des mots (ndir rap Lotfi Double Kanon Dangerous album). Après l'appariement entre les mots-clés et les concepts de l'ontologie en exploitant les synonymes, On obtient le vecteur de caractéristique ci-dessous.

Rap	Lotfi Double Kanon	Album	Dangerous
-----	--------------------	-------	-----------

Figure 15 : Vecteur de caractéristiques 1

En exploitant les propriétés d'objets et les relations sémantiques entre concepts et instances, on trouve que Rap est une instance du concept Genre, Lotfi Double Kanon est une instance of concept chanteur, Album est un concept et Dangerous est une instance du concept Album. On utilise la relation chante_theme entre Lotfi Double Kanon et l'instance Social et la relation has_theme entre l'instance Rap du concept genre est l'instance Social du concept thème. Alors, le thème de la chanson est social.

Maintenant, considérons un cas de désambiguïsation, l'extrait de chanson à classer est le suivant: "Mə bqāš Blasa fə Bladi Blad ntā3 l'banat, nruh blə rəj3a bəl visa u la hərəga" (je ne peux pas rester dans mon pays, Pays conduit par les femmes. Je pars sans retour, avec visa, sinon illégalement).

Après élimination des mots vides (Mə, fə, l', u, la), on obtient la liste des mots (bqāš Blasa Bladi Blad ntā3 banat, nruh blə rəj3a bəl visa a hərəga),

Après l'appariement entre les mots-clés et les concepts de l'ontologie en exploitant les synonymes, On obtient le vecteur de caractéristique ci dessous

Bladi	banat	visa	Haraga
-------	-------	------	--------

Figure 16 : Vecteur de caractéristiques 2

Partie II La catégorisation conceptuelle multi-langues de chansons

bladi (synonyme contextuel attaché au thème national); banat (synonyme contextuel attaché au thème Sentimental), Visa and Haraga (synonymes contextuels attachés au thème à l'immigration attaché au thème Social). La chanson appartient à l'un des thèmes suivant: National, sentimental ou Social. C'est un cas de désambiguïsation :

Le système demande à l'utilisateur d'introduire d'autres informations à propos de la chanson. Si l'utilisateur Ajoute « groupe MBS ». Le système trouve que groupe MBS est une instance du concept chanteur, en exploitant la relation chante_theme entre groupe MBS et l'instance Social du concept theme. Dans ce cas la classe de la chanson est sociale.

Dans le cas où l'utilisateur n'ajoute aucune information, le système suit le principe de Khan et calcule le score de chaque instance (National, sentimental ou Social). Il trouve que le score de l'instance social est le plus grand de ces scores, d'où le thème de la chanson est social.

II.3. Expérimentations

Pour tester notre approche, nous allons considérer le domaine de chansons algériennes, les chansons algériennes nous semblent être un bel exemple illustratif car elles présentent la particularité d'être chantées dans des langues différentes; l'exemple type est notre hymne footballistique qui est « One, two, three viva l'Algérie ». D'après notre étude des différents genres et styles de chansons algériennes, nous avons constaté qu'il y'a une relation étroite entre le genre de musique et le thème de la chanson et ainsi qu'entre le chanteur et le genre de chanson qu'il chante.

En plus d'après [98, 99, 100, 101, 102, 103, 104] les thèmes les plus récurrents dans les chansons algériennes sont: Célébration, hymne National, chant religieux, Sentimental, Social, Sportif et pour chaque thème on a déterminé la liste de mots-clés le décrivant.

Le but de l'étude est de mesurer l'efficacité de l'approche basée concept par rapport à l'approche de classification basée sur l'algorithme Bayes Naïf pour la classification de chansons décrit dans [41].

Partie II La catégorisation conceptuelle multi-langues de chansons

Tableau 10 : Classes des corpus

Classes	Nombre de chansons dans le corpus d'apprentissage	Nombre de chansons dans le corpus de test
Célébration	45	38
National	30	26
Religieux	58	52
Sentimental	120	112
Social	47	43
Sportif	60	55

Les résultats des classifications sont reportés dans les tableaux ci-dessous qui montrent les matrices de confusion pour le classificateur Bayes Naïf et la classification conceptuelle.

Tableau 11: Matrice de confusion avec classification conceptuelle

Classes	Célébration	National	Religieux	Sentimental	Social	Sportif	Total
Célébration	36	0	0	1	1	0	38
National	0	25	0	0	0	1	26
Religieux	1	0	49	2	0	0	52
Sentimental	2	0	0	108	2	0	112
Social	1	0	1	1	40	0	43
Sportif	2	2	1	1	2	47	55

Tableau 12: Matrice de confusion avec l'algorithme Bayes naïf

Classes	Célébration	National	Religieux	Sentimental	Social	Sportif	Total
Célébration	20	2	4	7	2	3	38
National	2	14	1	4	2	3	26
Religieux	7	1	30	9	5	0	52
Sentimental	15	7	4	60	9	17	112
Social	1	2	5	5	28	2	43
Sportif	6	6	5	10	2	26	55

Partie II La catégorisation conceptuelle multi-langues de chansons

Pour mesurer les performances de chacune des méthodes, nous utilisons la précision et le rappel ainsi que la mesure F1. Les figures suivantes nous renseignent sur le rappel et la précision ainsi que F1-mesure dans un cadre comparatif entre l'algorithme Naive Bayes et l'approche de classification basée ontologie.

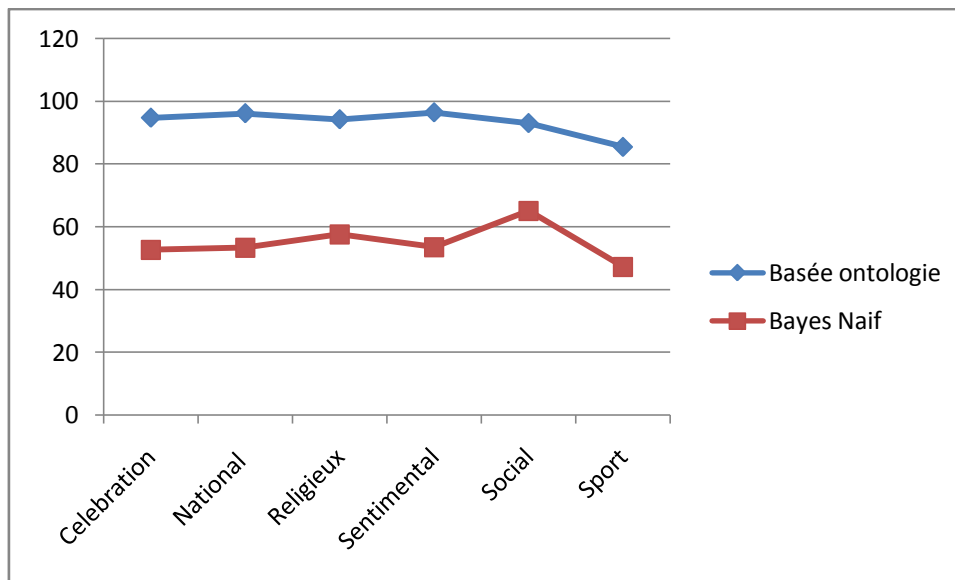


Figure 17: Rappel de la classification

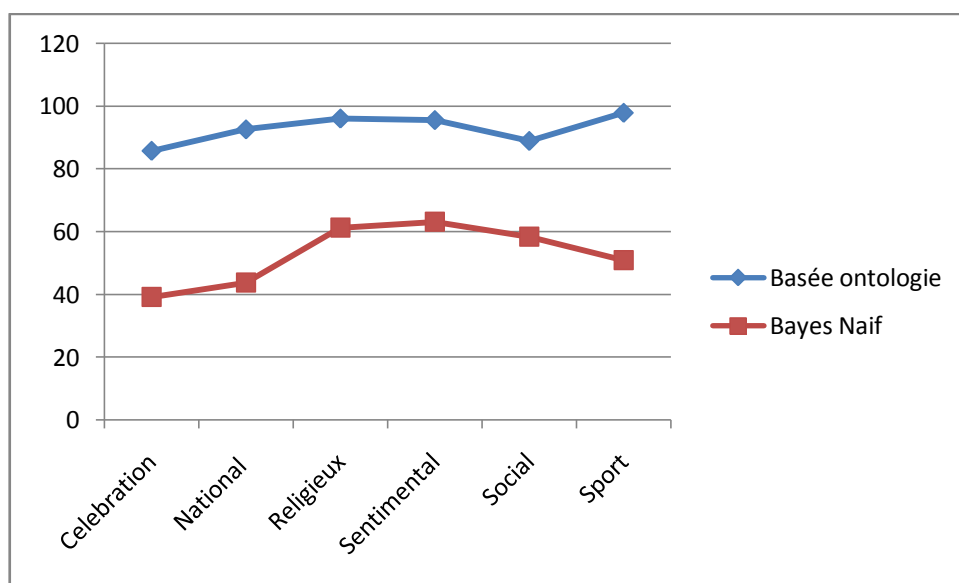


Figure 18: La précision de la classification

Partie II La catégorisation conceptuelle multi-langues de chansons

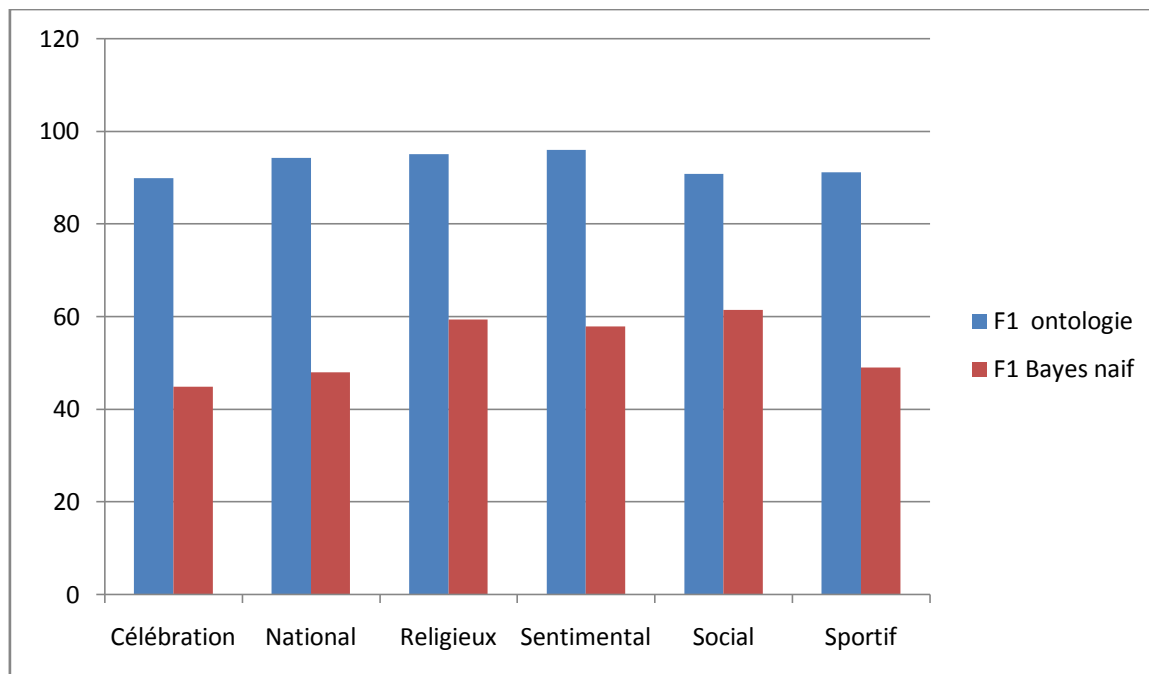


Figure 19: Histogramme montrant la F_1 mesure pour chaque catégorie pour les 2 algorithmes

L'historgramme (figure 19) montre que l'approche conceptuelle utilisant l'ontologie de domaine des chansons algérienne donne d'excellents résultats avec une F_1 mesure nettement supérieure à celle de Bayes naïf grâce à l'exploitation des liens: concept/terme(s) d'une part, et d'autre part, l'exploitation des relations sémantiques structurant l'ontologie, ainsi que la prise en compte des synonymes et des homonymes.

II.4. Conclusion

Dans cette partie, nous avons présenté une méthode de classification conceptuelle de chansons en utilisant une ontologie de domaine décrivant le domaine de chansons.

Pouvoir effectuer une indexation conceptuelle et une catégorisation sémantique de chansons en utilisant les ontologies est un axe de recherche très motivant. Dans ce cadre, nous avons proposé un modèle d'indexation à partir d'ontologie de domaine, puis nous avons développé un système de catégorisation de chanson selon le thème.

Un premier prototype concrétisant cette approche est développé qui permet l'indexation et la catégorisation conceptuelle de chansons.

L'évaluation de notre approche conceptuelle d'indexation et de catégorisation de chansons a donné de bons résultats par rapport au classificateur bayes naïf.

III. La recherche conceptuelle de chansons

III.1. Introduction

L'objectif de cette partie est d'indexer et rechercher une chanson en utilisant premièrement notre approche de recherche conceptuelle puis la recherche traditionnelle par mots-clés et de répondre à la question suivante:

- Quel est l'apport de l'indexation et la recherche basée concepts par rapport à une indexation et recherche classique par mots-clés ?

III.2. L'indexation et la recherche conceptuelle

Notre travail porte sur la facette bibliographique et la facette textuelle de l'information musicale (les chansons) et utilise l'ontologie créée précédemment dans la section 4 partie 1 de ce chapitre comme moyen de représentation des connaissances du domaine de chansons.

III.2.1. Conception générale du système d'indexation et de recherche de chansons

La conception de notre système d'indexation et de recherche conceptuelle de chanson est décrite comme suit:

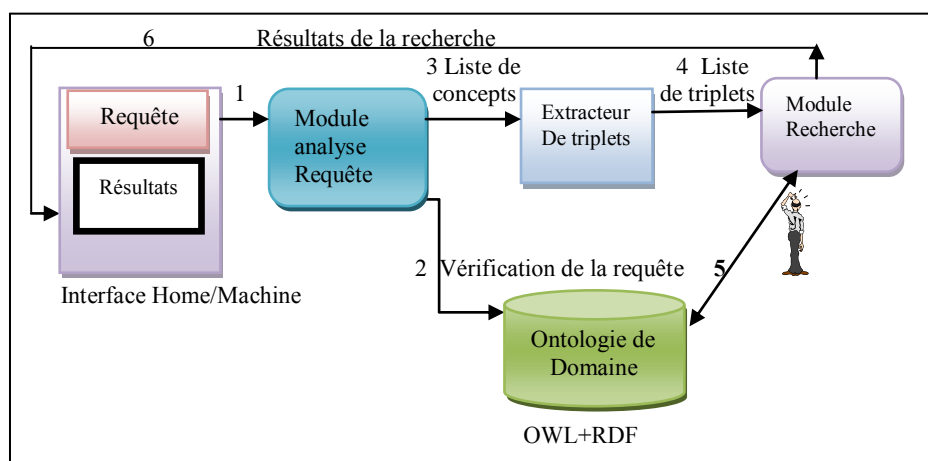


Figure 20: Conception du Module d'indexation et de Recherche.

III.2.1.1. Ontologie de domaine

Partie III La recherche conceptuelle de chansons

L'ontologie de domaine est l'ontologie créée dans la section 4 partie 1 de ce chapitre. Elle encapsule les entités et les synonymes. Les entités peuvent être des classes, des instances, des propriétés d'objets ou des propriétés de type de données. Pour chaque entité, une liste de synonymes qui lui est attachée est extraite.

Cette ontologie est utilisée dans la phase d'analyse linguistique des requêtes pour vérifier soit l'appartenance des termes initiaux de la requête à l'ontologie, soit l'appartenance de leurs synonymes dans le cas où ces termes n'appartiennent pas à l'ontologie. Elle est aussi utilisée dans la phase de la génération des requêtes SPARQL afin de vérifier s'il y a des relations sémantiques entre les différents termes de la requête, et d'enrichir la requête par toutes les relations trouvées.

III.2.1.2. Module Analyse

Le module d'analyse consiste à segmenter la requête entrée par l'utilisateur pour identifier ses différents composants. Puis, il supprime les mots vides et identifie chaque composant dans l'ontologie en exploitant les synonymes.

Afin d'illustrer l'analyse, nous allons examiner la requête suivante: "the song we are the champion". Le module d'analyse décompose la requête en six composantes: the, song, we, are, the, champion. Il supprime les mots (the, we, are, the) qui sont des mots vides appartenant à stop-list. Enfin notre requête contient, les mots (song, champion)

III.2.1.3. Extracteur de triplets

La traduction de la requête au langage SPARQL est nécessaire. Le langage de requêtes SPARQL a une structure à base de graphes et peut être construit en combinant les modèles de triplets extraites à partir de la requête.

Il est difficile de définir quels sont les sujets, les objets ou les prédicats alors on utilise l'ontologie du domaine pour les extraire. Le procédé d'extraction de triplets est le suivant:

L'extracteur de triplet identifie chaque composant dans l'ontologie en exploitant les synonymes. Si ni le terme, ni aucun de ses synonymes ne font pas partie de l'ontologie, ce terme sera automatiquement retiré de la liste des composants de la requête. Le résultat de cette phase est la séquence de concepts, des instances et des relations de l'ontologie qui appartiennent à la requête.

L'extracteur de triplets traduit la séquence précédente à un ensemble de triplets < sujet, prédicat, objet > où sujet et objet peuvent être des classes, des instances ou vide. Le prédicat peut être une propriété d'objet, une propriété de type de données, une relation de subsomption (is_a) ou vide.

Partie III La recherche conceptuelle de chansons

Donc, l'extracteur de triplet est utilisé pour identifier tous les triplets possibles d'une requête comme suit:

- Prendre les sujets deux à deux avec un prédicat vide; afin de trouver toutes les relations sémantiques qui peuvent exister entre les sujets de la requête, et d'enrichir la requête par ces relations trouvées.
- Prendre les sujets deux à deux avec tous les prédicats existants; afin de vérifier s'il existe des triplets valides dans la requête.
- Prendre chaque sujet avec tous les prédicats existants; afin de vérifier si le sujet est le domaine ou le range du prédicat et remplacer le sujet vide par une variable qui sera utilisé par la suite pour générer les chansons recherchées.

Continuons avec notre requête: "the song we are the champion". Après exécution du module analyse, la requête contient les mots (song, champion). Après correspondance avec les entités de l'ontologie de domaine, nous obtenons: song → classe / champion → instance après l'extraction du triplet de la requête, on obtient: "chanson, -, champion".

III.2.1.4. Module de recherche

La recherche a pour objectif d'accéder à l'ontologie de domaine pour renvoyer les chansons satisfaisant la requête. Ceci est réalisé par l'utilisation de notre algorithme de raffinement de requête décrit ci-dessous

Partie III La recherche conceptuelle de chansons

Algorithme de raffinement de requêtes

Entée: triplets de la requête. // résultat de l'extracteur de triplets

Sortie: Chansons retrouvées

Procédure:

Pour chaque triplet faire

Si le triplet contient seulement un seul terme, deux composants du triplet sont vides

Si le composant existant est une classe alors les individus appartenant à cette classe doivent être recherchés. // *par exemple: si on a le triplet <chanson,_,_> tous les individus de la classe chanson seront affichés //*

Si le composant existant est une propriété alors, nous devons rechercher les individus reliés par cette propriété // *par exemple: si on a le triplet <_,chante_> tous les individus reliés par la propriété chante seront affichés//*

sinon

1. Si la relation du triplet est vide

1.1. Si les deux composants sont des classes; le problème devient celui de rechercher une relation susceptible de relier les deux classes. Cette relation peut être une relation de subsumption ou une propriété d'objet.

Si la relation est une relation de subsumption alors

On restreint l'interrogation sur la classe la plus spécifique.

Sinon

la relation du triplet vide est remplacée par la propriété d'objet trouvée. // *par exemple: si on a le triplet <chanteur, _,chanson > devient <chanteur, chante, chanson //*

1.2. Si le triplet contient un des composants est une classe et l'autre est une instance.

Si l'instance appartient à cette classe.

On restreint l'interrogation sur l'instance. // *par exemple: si on a le triplet <chanson, _, champion > la chanson " we are the champion" sera affichée //*

Sinon

On doit rechercher s'il existe une propriété d'objet reliant les deux concepts, et on remplace la relation vide par la propriété d'objet trouvée // *par exemple: si on a le triplet <Queen, _,chanson > devient <Queen, chante, chanson > toutes les chansons du chanteur Queen seront affichées //*

1.3. Si les deux composants du triplet sont des instances.

On doit rechercher s'il y'a une propriété d'objet qui relie les deux instances alors

Remplacer la relation vide du triplet par la propriété d'objet trouvée. // *par exemple: si on a le triplet <Queen, _ , champion> devient <Queen, chante, champion> la chanson " we are the champion" sera affichée //*

1.4. Si l'une des composants est une instance et l'autre est un littéral.

On doit rechercher s'il existe une propriété de type de donnée dont le domaine est le type de l'instance et le range est le type du littéral. On remplace la relation vide par la propriété de type de donnée trouvée.

2. Si la relation n'est pas vide

Si l'une des composants est vide,

On doit vérifier si le composant qui existe est le domaine ou le co-domaine de la relation. On remplace le composant vide par une variable. // *par exemple: si on a le triplet <_,chante,chanson> devient <x, chante ,chanson > où x représente tout les chanteurs.*

Sinon

On doit vérifier si ce triplet est valable dans l'ontologie c'est-à-dire si cette relation relie les deux composants. S'il n'existe aucune relation sémantique entre les composants du triplet, ce dernier sera supprimé de la liste des triplets de la requête.

III.3. Implémentation de l'indexation et la recherche conceptuelle

III.3.1. La recherche conceptuelle

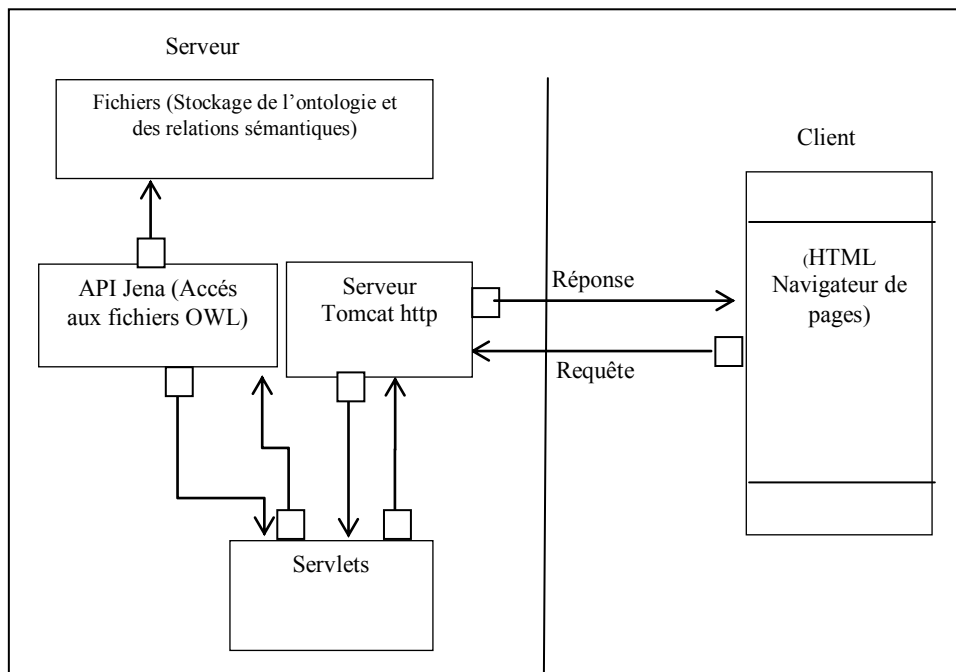


Figure 21: Architecture logicielle du système développé

Nous avons implémenté le système sous forme d'une application Web. Pour cela nous avons utilisé le langage Java et les Servlets qui permettent une grande flexibilité et la portabilité de l'application. Celle-ci rentre dans le cadre de la nouvelle génération du Web (le Web sémantique). Nous avons utilisé aussi les outils suivants: JDK7: Le Java Development Kit. Eclipse WTP: Web Tools Platform. Apache Tomcat 7: est un conteneur libre de servlets et JSP Java EE. Dans notre application deux options sont proposées à l'utilisateur. Dans le premier cas, l'utilisateur remplit un formulaire de requêtes pour effectuer une recherche simple en donnant seulement le titre de chansons ou une recherche avancée en donnant le nom du chanteur, le genre, la langue, la région, les instruments, le thème ou les mots de refrain de la chanson. SPARQL est utilisé pour explorer l'ontologie pour afficher les résultats. Une fois que les chansons sont trouvées, elles sont retournées à l'utilisateur. Ce dernier peut écouter ces chansons qui sont en format wav. L'utilisateur peut aussi indexer et insérer de nouvelles chansons. La figure 22 montre l'interface de recherche de chansons de l'application Web.

Partie III La recherche conceptuelle de chansons

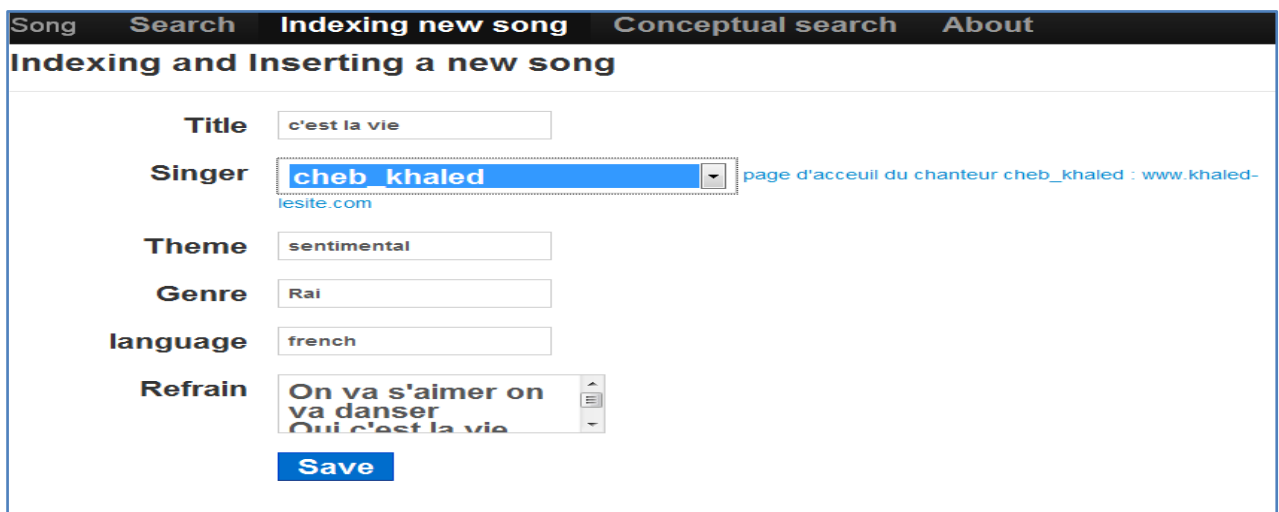
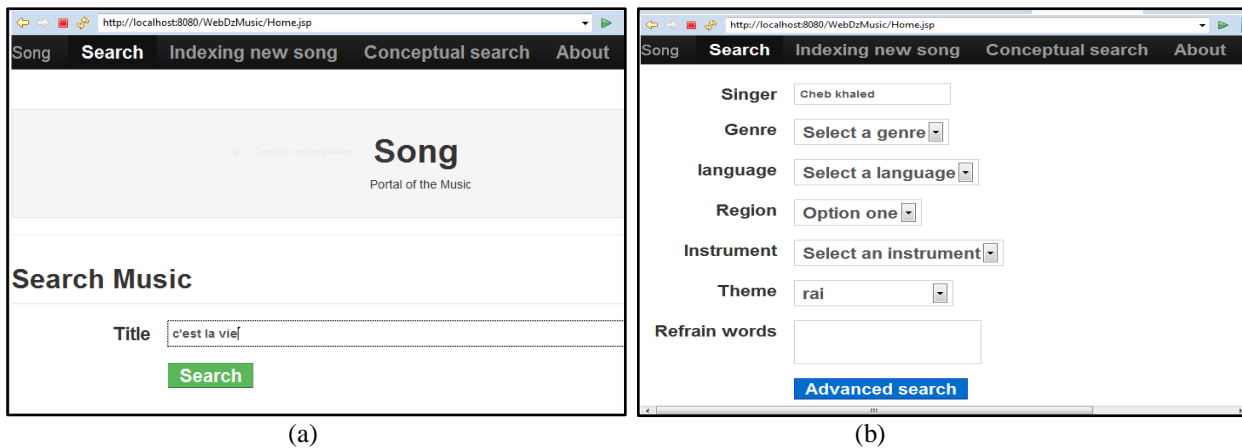


Figure 22: Interface de l'application de recherche web (a) : recherche simple, (b) recherche avancée, (c) indexation et insertion d'une nouvelle chanson

- Si le champ titre est rempli, le système retourne toutes les instances de chansons avec le même titre de tous les chanteurs. Sinon, l'utilisateur peut remplir d'autres champs.
- Si le champ chanteur est rempli, le système retourne toutes les chansons de ce chanteurs en exploitants la relation chanteur chante chanson. S'il n'ya pas de résultats, alors, le système effectue une expansion de la requête. Il performe une seconde requête qui retourne les chansons qui ont le genre du chanteur.
- La recherche portant sur les critères: instrument, région et langue sont utilisés pour enrichir la recherche, ils sont utilisés en conjonction avec le style de chansons ou chanteurs. Le résultat retourné est les chansons dont le genre est connu dans une région

Partie III La recherche conceptuelle de chansons

donnée ou les chansons pour un genre utilisant les instruments sélectionnés par le critère instrument ou le critère langue.

- La recherche sur le critère refrain retourne les chansons qui ont le même refrain ou leurs synonymes.
- La recherche sur le critère thème retourne les chansons avec le thème.
- L'utilisateur peut combiner plusieurs champs dans le processus de recherche. Le système utilise les relations sémantiques entre les concepts et les instances pour trouver les chansons qui correspondent aux critères de recherche.

Dans le cas où l'utilisateur ne connaît aucune information sur la chanson juste un extrait de parole il peut effectuer une recherche en langage naturel ceci correspond à notre deuxième technique. Dans ce cas on applique l'algorithme de raffinement de la requête.

La figure 23 montre l'interface de recherche conceptuelle web de chansons qui permet de saisir une requête en langage naturel.

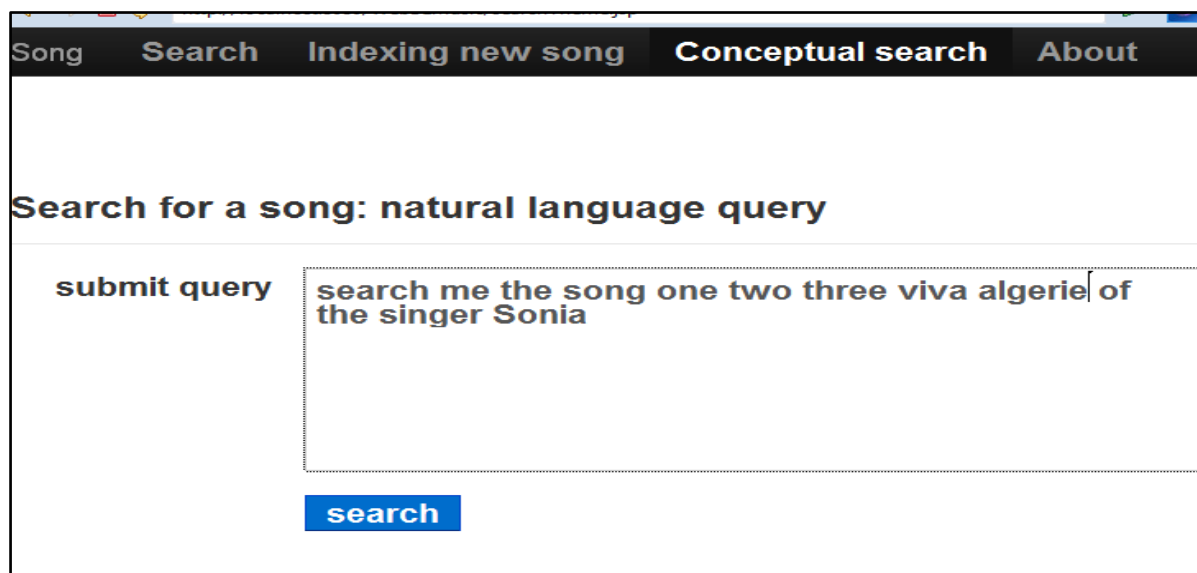


Figure 23: Interface de l'application de recherche conceptuelle Web

III.3.2. Approche d'indexation et de recherche par mots-clés

Dans ce cadre nous avons créé une base de données classique de chansons contenant plusieurs tables: table de chansons, table de chanteurs, table de genres, table de thèmes et table d'instruments. Nous avons utilisé la recherche en texte intégral de SQL Server (sql full text search) qui permet d'exécuter des requêtes de textes intégral sur des données caractères dans des tables SQL Server. Nous l'avons appliqué au titre, au refrain et à la parole. Le résultat de la recherche est

Partie III La recherche conceptuelle de chansons

l’affichage de toutes les chansons contenant les mots de la requête dans le titre si la recherche porte sur le titre, dans le refrain si la requête porte sur le champ refrain ou dans la parole dans le cas où on effectue une recherche sur la parole. L’utilisateur peut écouter les chansons trouvées. Les chansons sont sauvegardées dans la table chanson et elles sont lues en utilisant MPlayer. L’application qui recherche les chansons par mots-clés est réalisée en utilisant C# 2010 et sql server 2008.

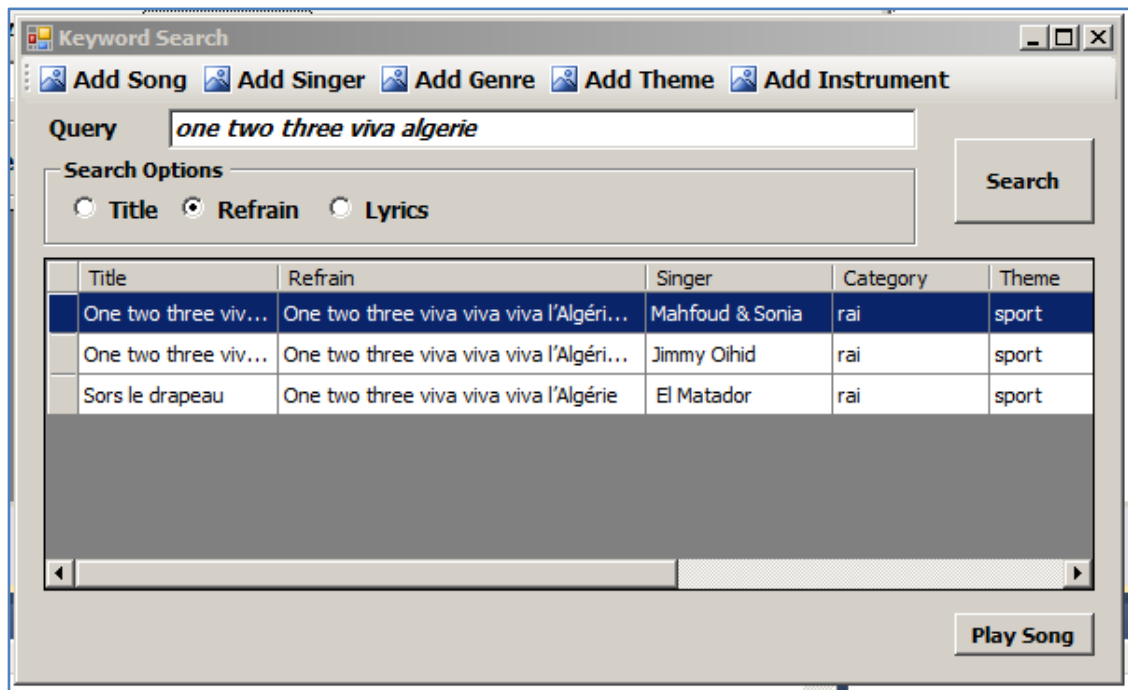


Figure 24: Résultats de la recherche pour la chanson “One Two Three Viva l’Algérie”

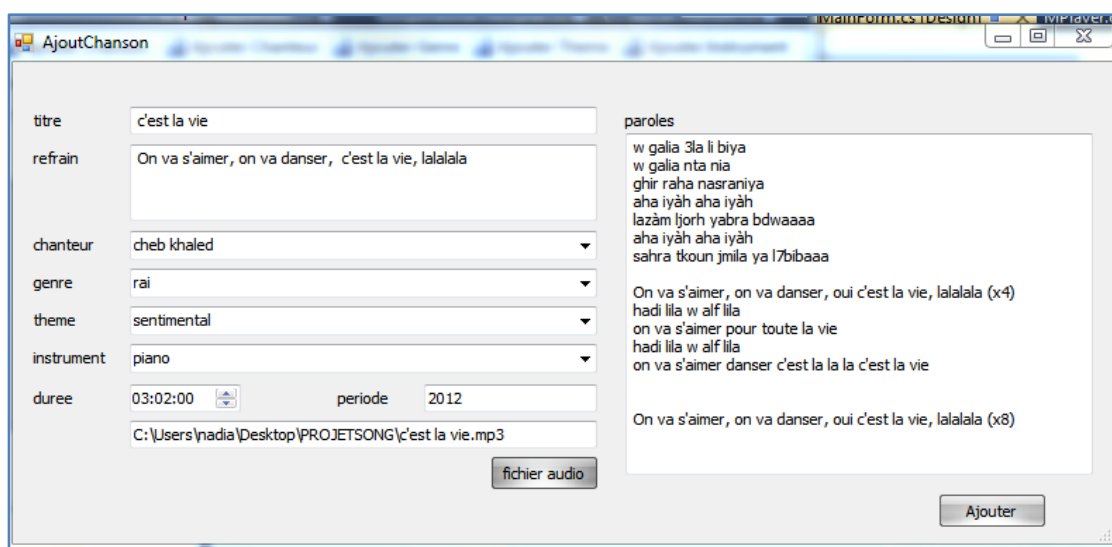


Figure 25: Ajout de chansons

Partie III La recherche conceptuelle de chansons

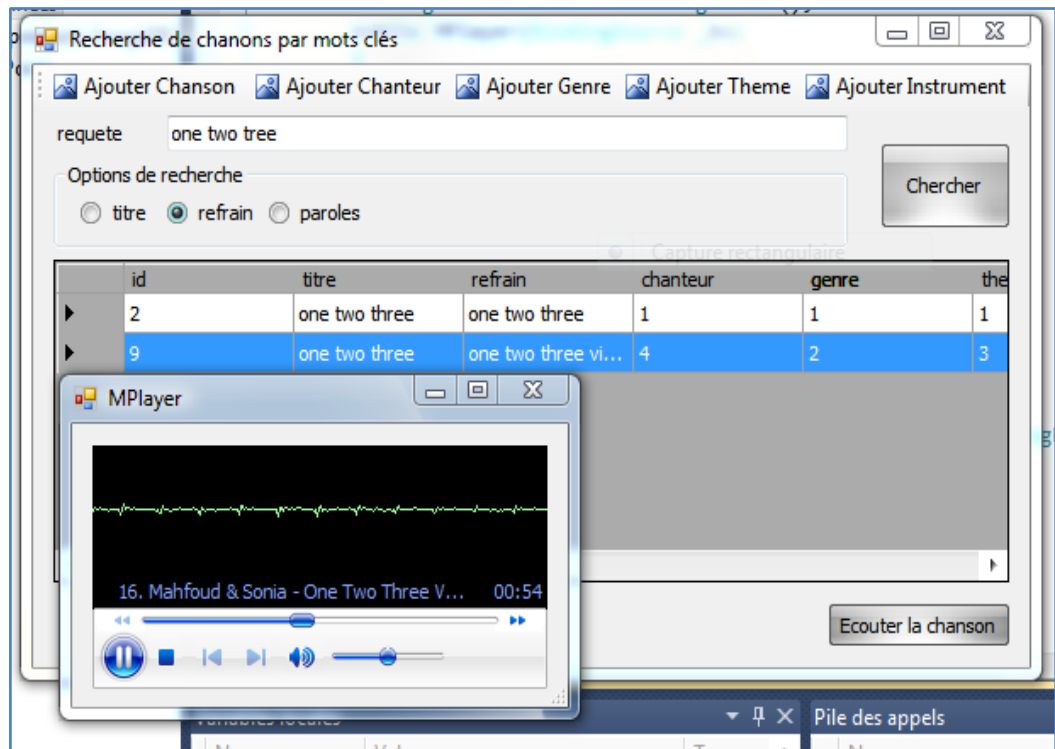


Figure 26: Résultat de recherche pour la chanson *one two three viva l'Algérie* et écoute de la chanson

III.4. Evaluation de l'approche

Dans ce paragraphe, nous exposons les évaluations qui ont été menées dans le but d'étudier le comportement et la performance de notre approche ainsi que les résultats obtenus et nous détaillons un exemple illustratif de recherche.

III.4.1. Mesures d'évaluation

Le but de l'évaluation est de mesurer l'efficacité de l'approche basée concept par rapport à l'approche classique basée mots-clés. Plus spécialement, notre contribution proposée dans l'approche est d'évaluée:

- Quel est l'apport de l'indexation basée concepts par rapport à une indexation classique?

Pour évaluer l'approche, des d'expérimentations ont été montées. Premièrement basé sur l'indexation classique ensuite les expérimentations basées sur l'indexation conceptuelle (en utilisant l'ontologie de domaine).

Partie III La recherche conceptuelle de chansons

On dispose

- d'une collection de chansons de références
- d'un ensemble de requêtes (15 requêtes)
- avec pour chacune, l'ensemble des chansons pertinentes donner par un juge

Les niveaux de rappel ne sont pas unifiés pour l'ensemble des requêtes, D'après la littérature on retient 11 points de rappel standards: de 0.0 à 1.0 à pas de 0.1.

Les calculs « à x réponses » permettent d'obtenir des valeurs de précision pour une valeur de rappel donnée, rappel constant tant que l'on ajoute des réponses non pertinentes ce qui donne une précision décroissante au sein d'une même valeur de rappel.

On se ramène à une fonction $f(r) = p$ en utilisant la règle d'interpolation :

$$P_{\text{interp}}(r) = \max (P (r')) \text{ pour } r' \geq r$$

On calcule la précision moyenne pour les 15 requêtes aux différents points de rappel pour les 2 méthodes (mots-clés et conceptuelle), On obtient les résultats mentionnés dans le tableau 13.

Tableau 13: Précisions aux points de rappels pour les deux méthodes

Point de rappel	Précision	
	L'approche basée mots-clés	L'approche conceptuelle
A 0.00	0,6751	0,8520
A 0.10	0,5459	0,7543
A 0.20	0,4427	0,6743
A 0.30	0,3786	0,6292
A 0.40	0,3104	0,5537
A 0.50	0,2753	0,5038
A 0.60	0,2208	0,4179
A 0.70	0,1866	0,3615
A 0.80	0,1179	0,2864
A 0.90	0,0551	0,2187
A 1.00	0,0188	0,1712

Partie III La recherche conceptuelle de chansons

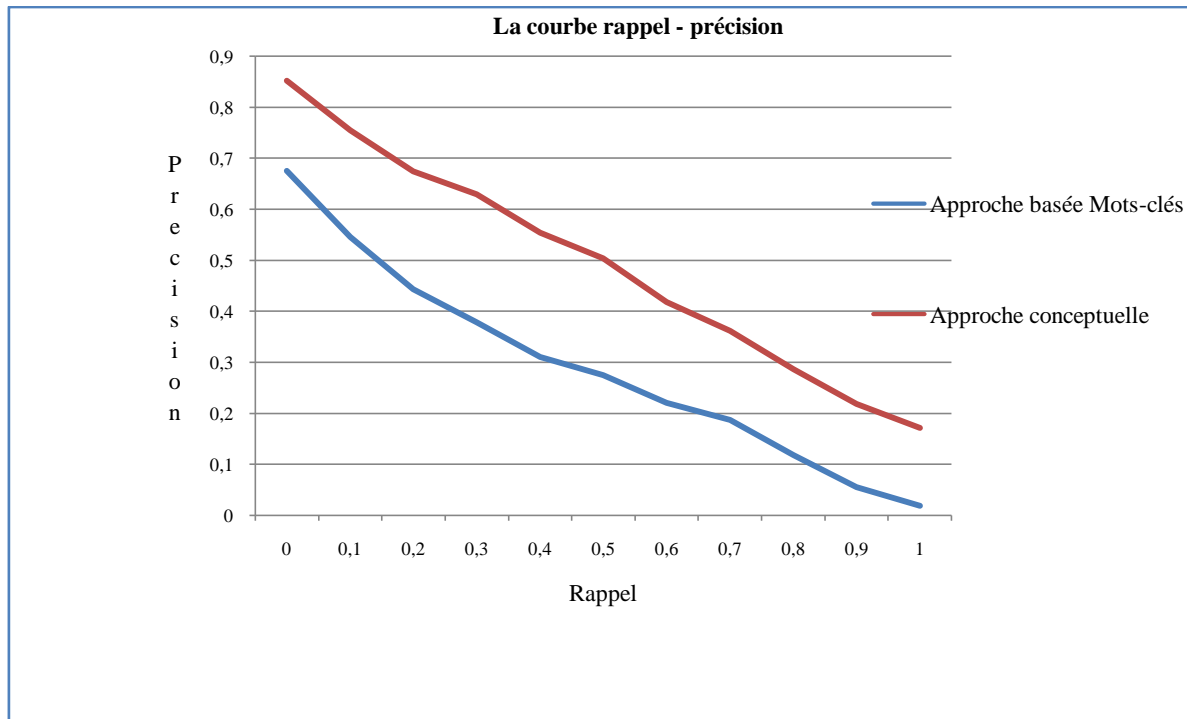


Figure 27: La courbe rappel-précision

III.4.2. Discussion

Les résultats obtenus sont bons. L'analyse globale des résultats (comme le montre la courbe rappel-précision de la figure 27, donne des résultats comparables à ceux obtenus par une méthode de recherche classique par mots-clés: liste de mots simples (bag of words).

La figure 27 montre que la courbe de l'approche basée concepts dépasse (elle se situe en haut à droite de) la courbe de l'approche basée mots-clés donc l'approche basée concept est la meilleure.

L'approche utilisant mots-clés est connue comme générant beaucoup de bruit.

III.4.3. Exemple d'illustration

III.4.3.1. Exemple Recherche conceptuelle en remplissant le formulaire

Si l'utilisateur remplit tous les champs « Interface de l'application webDz-Song Recherche d'une chanson ». Il y'aura une seule réponse comme résultat.

Partie III La recherche conceptuelle de chansons

The screenshot shows the Dz-Music website's search interface. At the top, there are navigation links: "Dz-Music", "Recherche Chanson بحث", "Nouvelle Chanson أغانى جديدة", "Identification de Thème تحديد الموضوع", and "A propos فيما يخص". A search bar contains the text "aïcha" and a green "Recherche" button. Below the search bar, there is a "Recherche Avancée" section with a "Capture rectangulaire" watermark. This section includes several filters: "Chanteur" (cheb_khaled), "Genre" (Rai), "Langue" (selectionné une langue), "Région" (Option one), "Instrument" (selectionner un instrument), and "Thème" (Sentimental). There is also a "Mots Refrain" dropdown menu showing "Aïcha, Aïcha écoute-moi, Aïcha, Aïcha t'en vas pas". A blue "Recherche Avancée" button is at the bottom of the filters.

Figure 28: Exemple de recherche de la chanson Aïcha

The screenshot shows the search results page on Dz-Music. The header is dark with the Dz-Music logo and navigation links. Below the header, there is a table with the following columns: "Titre", "Thème", "Langue", "Chanteur", "Genre", "Refrain", and "Ecouter". The table contains one row of results for the song "aïcha" by "cheb_khaled". The "Ecouter" column has a green play button icon. Below the table, there are navigation buttons: "Prev", "1", "2", "3", and "Next".

Titre	Thème	Langue	Chanteur	Genre	Refrain	Ecouter
aïcha	Sentimental	français	cheb_khaled	rai	aïcha	

Figure 29: Réponse du système pour la recherche de la chanson Aïcha

L'utilisateur peut écouter la chanson en cliquant sur écouter

Si l'utilisateur fait entrer le nom du chanteur seulement, il aura l'affichage de tous les chansons (instances de l'ontologie) chanter par ce chanteur avec affichage 10 chansons par page et gestion dynamique de page (page précédente/ page suivante).

III.4.3.2. Exemple Recherche conceptuelle en remplissant la requête en langage naturel

Partie III La recherche conceptuelle de chansons

The screenshot shows the top navigation bar of the Dz-Music website with links for 'Recherche' (بَحْث), 'indexation Nouvelle Chanson' (أَغْنِيَة جَدِيدَة), 'Recherche libre' (بَحْث حُر), and 'A propos' (فِيمَا يَخُص). Below the navigation bar, the main heading reads 'recherche d'une chanson requête en langage naturel'. A search input field contains the text 'les chansons de lotfi de thème social', and a blue button labeled 'rechercher' is positioned below it.

Figure 30: Recherche d'une requête exprimée en langage naturel

- 1- L'utilisateur entre la requête : «les chansons de lotfi de thème social »
- 2- Le décomposeur de requêtes décompose la requête on sept composants:
les /chansons/de /lotfi /de /thème/social
- 3- Après la correspondance avec les entités de l'ontologie chanson on a :
Les → mot vide₁
Chansons → classe₁
De → mot vide₂
Lotfi → instance₁
De → mot vide₃
Thème → classe₂
Social → instance₂
- 4- Après l'extraction de tous les triplets possibles de la requête on obtient :

Tableau 14 : Les triplets de la requête

Triplet	Type de triplet
chansons, -, lotfi	classe, vide, instance
chanson, -, thème	classe, vide, classe
chanson, -, social	classe, vide, instance
lotfi, -, thème	instance, vide, classe
lotfi,-, social	instance, vide, instance
thème, -, social	classe, vide, instance

Partie III La recherche conceptuelle de chansons

III.5. Conclusion

Nous avons présenté dans cette partie un algorithme de raffinement de requêtes dans un système de recherche conceptuelle de chansons. L'approche que nous avons présentée utilise une ontologie de domaine pour la représentation et l'indexation de chansons

Les résultats obtenus sont bons et comparables à ceux obtenus par une méthode de recherche classique par mots-clés.

Conclusion générale

La notion d'information musicale, au travers du langage sur lequel elle s'appuie, est complexe et recèle de multiples facettes qui sont autant d'axes et d'objets de recherche possibles. Cependant, elle est fortement tributaire de la perception que nous pouvons en avoir selon l'aire géographique et la culture à laquelle nous appartenons. De plus, l'utilisation du langage musical demeure aujourd'hui instable, sans cesse mouvante et sans limites. Mais cette complexité en montre par ailleurs toute la richesse.

A ce titre, proposer des méthodes et outils de catégorisation et de recherche répondant le plus globalement aux besoins des utilisateurs possibles s'avère ardu, à cause de la multiplicité de ces facettes. Des difficultés s'opposent à cette démarche comme, par exemple, celle d'évaluer avec précision l'objet recherché. Différentes méthodes et applications fonctionnelles existent mais aucune apparemment ne tend à satisfaire pour le moment un objectif qui semble primordial au vu de la diversité des aspects que peut revêtir l'information musicale: pouvoir effectuer une catégorisation et une recherche de chansons.

Dans cette thèse, nous nous sommes intéressés à proposer des méthodes de catégorisation de chansons.

Dans ce cadre là, nous avons présenté principalement trois contributions dans les processus d'indexations de chansons à savoir: l'utilisation de l'algorithme Bayes naïf, l'adaptation d'un algorithme de colonies de fourmis et la création et l'utilisation d'une ontologie de domaine pour la représentation du domaine de chansons dans les processus d'indexation d'une collection de chansons.

La première contribution consiste en l'application du classificateur Bayésien naïf pour la catégorisation de chansons, dans lequel, nous avons utilisé, en plus du titre et du refrain de la chanson, les caractéristiques de l'artiste ce qui a permis d'améliorer les performances du classificateur. Nous avons utilisé le classificateur Bayésien naïf comme un classificateur de références pour les deux contributions suivantes.

La deuxième contribution consiste en la proposition d'une approche originale qui réside dans le fait d'adapter un algorithme de colonies de fourmis pour identifier le thème d'une chanson afin de la classer. Dans cette approche, une classe de fourmis portant le même nom de catégorie correspond à une catégorie cible qui est déterminée par un mécanisme de classification. La distance entre les villes peut être considérée comme la similarité entre les catégories et les nœuds. Les phéromones correspondent à des poids de classification des nœuds. Chaque nœud du graphe

est un terme d'une chanson et les liens sont créés par les mesures de similarités entre les termes des chansons. Un algorithme est proposé qui consiste à diviser les fourmis en plusieurs groupes (le nombre de groupes est égale au nombre de catégories). A chaque cycle de l'algorithme, chaque groupe de fourmis construit un sous-ensemble de nœuds couverts. En partant d'un sous-ensemble I_k vide, les fourmis ajoutent à chaque itération un nœud à I_k choisi parmi l'ensemble des nœuds non encore sélectionnés. Les fourmis de chaque groupe traversent tous les nœuds du graphe. Le nombre de fourmis dans un groupe détermine le nombre d'itérations de ce groupe. Un chemin I_k de classe k qui peut décrire l'optimum de cette classe sera généré après que les fourmis du groupe k ont visités tous les nœuds. Le résultat de la classification sortira en comparant les concentrations de phéromones b des chemins I_k après toutes les itérations des fourmis de tous les groupes. La classification k décrite par le chemin I_k qui a la concentration de phéromone max est la classe de la chanson.

La comparaison des résultats a montré la supériorité de la démarche de l'indexation de chansons en utilisant l'algorithme de colonies de fourmis par rapport à celle utilisant le classificateur Bayésien naïf.

La troisième contribution consiste à créer une ontologie de domaine qui représente un axe de recherche très motivant pour représenter le domaine de chansons et utiliser cette ontologie pour:

- proposer un modèle d'indexation conceptuelle multi-langues de chansons
- développer un système de recherche conceptuelle de chansons en proposant un algorithme de raffinement de requêtes.

L'approche présentée a les avantages suivants

- ✓ Permettre une représentation fidèle du contenu sémantique des chansons;
- ✓ Faciliter la recherche d'information musicale au sein de collections hétérogènes en indexant tout type de chansons à partir des mêmes concepts;
- ✓ Permettre une recherche intelligente, et cela en exploitant la sémantique des liens entre concepts d'indexation.

L'approche présentée apporte une certaine intelligence dans le processus de recherche d'information musicale mais aussi plus de pertinence grâce à l'utilisation de l'ontologie de domaine comme technique d'indexation et par l'exploitation des liens sémantiques entre chansons. Un premier prototype concrétisant cette approche est développé qui permet l'indexation, la recherche de chansons.

L'évaluation de notre approche d'indexation conceptuelle de chansons a donnée de bons résultats comparés à la catégorisation de chansons utilisant le classificateur Bayésien naïf.

L'évaluation de notre approche de recherche conceptuelle de chansons a montré la supériorité de cette démarche par rapport à une représentation classique par mots-clés utilisée comme référence.

Perspectives

Les perspectives envisageables à nos travaux portent principalement sur quatre volets:

Un premier volet porte sur l'élargissement de la validation et de l'évaluation de nos approches en utilisant les bases de données publiques de musique telle que CAL 500 et la base de données Million chansons "Million Song Dataset".

Un deuxième volet consiste à enrichir le domaine d'application en ajoutant aux facettes textuelles et bibliographiques (étudiés dans notre cas) d'autres facettes (tonale, temporelle, harmonique, timbrale, éditoriale) et à évaluer cette approche par rapport aux folksonomies en utilisant les tags des utilisateurs dans les sites de chansons.

Un troisième volet porte sur une étude plus approfondie de la pondération des concepts de l'ontologie. En effet, jusqu'à présent dans les approches d'indexation classiques, si l'on peut noter un consensus assez large autour de la méthode de pondération en TF.IDF, un grand travail reste à faire concernant la pondération des concepts.

Un quatrième volet porte sur la combinaison des méthodes de catégorisation conceptuelle et celle utilisant l'algorithme de colonies de fourmis pour tirer profit des avantages qu'offre chacune d'elles. Ce volet utilise l'algorithme de colonies de fourmis pour la mise en œuvre d'un processus de désambiguïsation qui permet l'exploitation efficace de notre structure de l'ontologie de domaine. Il permet de calculer la distance entre deux concepts C_i et C_j , L'algorithme garantit le choix du concept le plus proche de la requête.

Publications

Publications dans des journaux internationaux (avec comité de lecture)

1. Lachtar Nadia and Bahi Halima. Categorization of songs using metadata, I3 journal Special Issue on Information Mining and Retrieval on the Web 2012 volume 12 Issue 2, pp. 105 pub.date November (2012)

Conférences internationales (avec comité de lecture)

1. Lachtar Nadia and Bahi Halima. Application of an ant colony algorithm for text indexing, in proceeding of international conference on multimedia computing and systems ICMCS'11 Ouarzazate MAROC du 7 au 8 avril 2011, lien du papier: <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=5931355>
2. Lachtar Nadia and Bahi Halima. Application of an ant colony algorithm for song categorizing using metadata, in proceeding of the International Conference on knowledge Discovery and Informations Retrieval KDIR'11, Paris, France du 26 au 29 Octobre 2011, lien du papier : <http://www.scitepress.org/DigitalLibrary>
3. Lachtar Nadia and Bahi Halima. Classification of a song, in proceeding of the 3^e Conférence internationale sur l'informatique et ses applications 2011 (CIIA 2011), Saida, Algérie, 13 au 15 Décembre 2011 le papier a été ajouté au DBLP. <http://www.informatik.uni-trier.de/~ley/db/conf/ciia/ciia2011.html>
4. Lachtar Nadia and Bahi Halima. Design and implementation of information retrieval system based ontology, in proceeding de l'international conference on Multimedia Computing and Systems (ICMCS 2014) Marrakech Maroc du 14 au 16 avril 2014, lien du papier : <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6900112>
5. Lachtar Nadia and Bahi Halima. Document search based ontology, in proceeding of the internationale conférence on Information Systems and Technologies ICIST'2015, Istanbul du 21 au 23 mars 2015.

Conférences internationales (avec comité de lecture)

1. Lachtar Nadia and Bahi Halima. l'identification de thèmes en indexation audio dans le cadre des Journées Ecole Doctorales JED 2010, les 14- 15 Juin Annaba.
2. Lachtar Nadia and Bahi Halima. Application d'un algorithme de colonie de fourmis pour la catégorisation de chansons dans le cadre international 1^{eres} journées Doctorales d'informatique D'Oran JDLIO'11 Oran, Algérie du 31 mai au 01 juin 2011.

Références bibliographiques

- [1] Gauvain J. L., Lamel L. and Adda G., “audio partitionning and transcription for broadcast data indexation”, in proceedings of cbmi 99, pp. 67-73, Toulouse, France, 1999.
- [2] Saunders J., “Real-time discrimination of broadcast speech/music”, in proceedings of icassp 96, pp. 993-996, Atlanta, Georgia, United States, 1996.
- [3] Scheirer E. and Slaney M., “Construction and evaluation of a robust multifeatures speech/music discriminator”, in proceedings of icassp 97, pp. 1331-1334, Munich, Germany, 1997.
- [4] Parris E. S., Carey M. J. and Harvey L. T., “Feature fusion for music detection”, in Proceedings of eurospeech 99, pp. 2191- 2194, Budapest, Hungary, 1999.
- [5] Carey M. J., Parris E. S. and Harvey L. T., “A comparison of feature for speech, music discrimination”, in Proceedings of icassp99, pp. 149- 152, Phoenix, Arizona, United States, 1999.
- [6] Seck M. , Bimbot F., Zudaj D. and Delyon B., “Two-class segmentation for speech/music detection in audio tracks”, in Proceedings of eurospeech99, pp. 2801-2804, Budapest, Hungary, 1999.
- [7] Seck M. , Magrin-Chagnolleau I. and Bimbot F., “ Experiments on speech tracking in audio documents using Gaussian mixture modeling”, in Proceedings of ICASSP 01, pp. 601-604, Salt lake city, Utah, United States, 2001.
- [8] Zissman M. A., “Conception of four approaches automatic language identification for telephone speech”, IEEE Transcription on speech and audio Processing, Vol. 4 n°1, pp. 31-44, January 1996.
- [9] Dutat M., Magrin-Changnolleau I. and Bimbot F., “Language recognition using time-frequency principal component analysis and acoustic modeling”, in Proceedings of ICSLP 00, Beijing Chine, 2000.
- [10] Dutat M., Magrin-Changnolleau I. and Bimbot F., “Acoustic modeling of spoken languages using time-frequency principal component analysis and hidden markov models: Application to language identification”, Submitted to IEEE Transactions on Speech and Audio Processing.
- [11] Magrin-Changnolleau I. and Parlangeau-Vallès N., “Audio indexing; what has been accomplished and the road ahead”, in proceedings of Sixth International Joint Conference on Information Sciences, Durham, North Carolina, United States, pp.911-914, 2002.
- [12] Kurimo M., “Indexing Audio documents by using Latent Semantic Analysis and SOM”, Erkki Oja and Samuel Kaski, editors, kohonen Maps, pp. 363- 374, Elsevier, 1999.

- [13] Sidhom S., “ Plate-forme d’analyse morpho-syntaxique pour l’indexation automatique et la recherche d’information de l’écrit vers la gestion des commerce ”, thèse de doctorat, Mars 2002.
- [14] Adaikkalavan R., Elkalifa L., Aslandogan Y. A., “topic identification through ontology-based concept generalization”, rapport technique, 2003.
- [15] Bigi B. and Smaili K., “ Identification thématique hiérarchique : Application aux forums de discussions”, Conférence traitement Automatique du Langage Naturel TALN 2002, Nancy, 24-27 juin 2002.
- [16] Downie J. S., “The Scientific Evaluation of Music Information Retrieval Systems: Foundations and Future”, *Computer Music Journal*, Vol. 28, N° 2, pp. 12-23. 2004.
- [17] Downie J. S., “Music Information Retrieval: Annual Review of Information”, *Science and Technology*, Vol. 37, pp. 295-340, 2003.
- [18] Aucouturier J. J., and Pachet F., “Representing musical genre: a state of the Art”, *Journal of New Music Research*, 32(1), pages 83-93, 2003
- [19] Byrd D., “The Fundamental Theorem of Music Informatics (maybe)”, Repéré à <http://www.informatics.indiana.edu/donbyrd/teach/FundamentalTheorem.html>, 2008
- [20] Mahedero J. P. G, Martinez A., Cano P., Koppenberger M. and Gouyon F., “Natural language processing of lyrics”, in *Proceedings of the 13th ACM International Conference on Multimedia*, pp. 475–478, New York, NY, USA, ACM Press, 2005.
- [21] Eleanor S., “What Motivates a Musical Query?”, in *Proceedings of the 1st International Conference on Music Information Retrieval (ISMIR’00)*, September, Plymouth, Massachusetts, USA, October 2000.
- [22] Howard J. and Schlichte J., “Repertoire international des sources musicales (RISM)”, in *Directory of computer assisted research in musicology*, pp. 11–24, 1988.
- [23] Temperley N., “The problem of definitive identification in the indexing of hymn tunes”, in *Foundations of music bibliography*, pp.227–239, 1993.
- [24] Dannenberg R. B. and Hu N., “Understanding search performance in query-by-humming systems”, in *proceedings of the 5th International Society for Music Information Retrieval Conference ISMIR*, pp. 232-237, 2004.
- [25] Goto M. and Hirata K., “Recent studies on music information processing”, in *Acoustical Science and Technology*, Vol. 25, N° 4, pp.419–425, 2004.
- [26] Phiwma N. and Sanguansat P., “An Improved Feature Extraction and Combination of Multiple Classifiers for Query-by-Humming”, in the *International Arab Journal of Information Technology*, Vol. 11, N° 1, 2014.

- [27] Li T. and Ogihana M., “Content-based music similarity search and emotion detection”, 2004.
- [28] Chua B. Y. and Lu G., “determination of perceptual tempo of music”, in CMMR 2004.
- [29] Liu D., Lu L. and Zang H-J., “Automatic mood detection from acoustic music data”, 2003.
- [30] Panagakis Y. and Kotropoulos C., “Music structure analysis regression of beat-synchronous audio features”, in Proceedings of the 13rd International Conference on Music Information Retrieval (ISMIR’12), pp. 271–276, Porto, Portugal, October 2012.
- [31] Schmidt E.M., Scott J. and Youngmoo E.K., “Feature learning in dynamic environment: modeling the acoustic structure of musical emotion”, in proceedings of the 13rd International Society for Music Information Retrieval Conference ISMIR, pp. 325–330, 2012.
- [32] Maazouzi F. and Bahi H., “Type-2 Fuzzy Gaussian Mixture Models for Singing Voice Classification in Commercial Music Production”, in International Journal of Signal and Imaging Systems engineering, Vol. 6, N°2, pp. 111– 118, 2013.
- [33] Purgina M., Kuznetsov A. and Pyshkin E., “Leveraging User Experience through Input Style Transformation to Improve Access to Music Search Services”, in journal Informatica : An International Journal of Computing and Informatics, Vol.38, pp. 11–19, 2014.
- [34] Logan B., Kositsky A. and Moreno, P., “Semantic Analysis of Song Lyrics”, in Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), Vol.2, pp. 827–830, Baltimore, Maryland, USA, 2004.
- [35] Knees P., Schedl M. and Widmer G., “ Multiple Lyrics Alignment: Automatic Retrieval of Song Lyrics”, in proceedings of the 6th International Society for Music Information Retrieval Conference ISMIR, pp. 564-569, 2005.
- [36] Brochu E. and Freitas N. D., “ Name that song! : A probabilistic approach to querying on music and text”, in proceedings of Advances Neural Information Processing Systems, pp. 1505-1512, 2002.
- [37] Besson M., Faïta F., Peretz I., Bonnel A. M. and Requin J., “Singing in the brain: Independence of Lyrics and Tunes”, in Psychological Science, Vol. 9, N°6, pp. 494-498, 1998.
- [38] Baumann S., “Using Cultural Metadata for Artist Recommendation”, in Proceeding of the 3rd International conference on Web Delivering of Music, pp. 138- 141, 2003.
- [39] Bainbridge D., Cunningham S. and Downie J., “Analysis of queries to a Wizard-of-Oz MIR system: Challenging assumptions about what people really want”, in proceedings of the 4th International Society for Music Information Retrieval Conference ISMIR, 2003.

- [40] Kleedorfer F., Knees P. and Pohle. T., "Oh Oh Oh Whoah! Towards Automatic Topic Detection in Song Lyrics", in proceedings of the 9th International Conference on Music Information Retrieval (ISMIR), pp. 287–292, September 2008.
- [41] Lachtar N. and Bahi H., "Categorization of songs using metadata" I3 journal Special Issue on Information Mining and Retrieval on the Web 2012 , Vol.12, Issue 2, pp. 105- 127, 2012.
- [42] Mayer R., Neumayer R. and Rauber A. "Rhyme and Style Features for Musical Genre Classification by Song Lyrics", in proceedings of the 9th International Conference on Music Information Retrieval ISMIR 2008.
- [43] Knees P., Pohle T., Schedl M. and Widmer G., "A music search engine Built upon Audio-based and Web-based similarity measures", in SIGIR, 2000.
- [44] Thanh D.T. and Shirai K., "Machine learning approaches for mood classification of songs toward music search engine", in proceedings of International conference of Knowledge and Systems Engineering, pp.144,149, 2009.
- [45] Li T. and Ogihara M., "Music artist style identification by semi-supervised learning from both lyrics and content", in proceedings of the 12th annual ACM international conference on Multimedia, pp. 364–367, New York, NY, USA, 2004.
- [46] Wei B., Zhang C. and Ogihara M., "Keyword generation for lyrics", in Proceedings of the Eighth International Conference on Music Information Retrieval, pp. 121–122, 2007.
- [47] Raimond Y., Samer A., Sandler M. and Giasson F., "The music ontology", in Proceedings of the 8^{eme} International Conference on Music Information Retrieval, pp. 417–422, 2007.
- [48] Kolozali S., Barthet M., Fazekas G. and Sandler M. "Knowledge representation issues in musical instrument ontology design", in Proceedings of the 12^{eme} International Conference on Music Information Retrieval ISMIR'11, pp. 465–470, Miami,Florida USA, October 2011.
- [49] Jeff Z.P., Taylor S. and Thomas E., "MusicMash2: Mashing Linked Music Data via An OWL DL Web Ontology", in journal.webscience.org. 2009.
- [50] Suárez-Figueroa M. C., Ateazing G. A. and Corcho O., " The landscape of multimedia ontologies in the last decade".in Multimedia Tools Appl. Vol. 6 issue 2 pp. 377-399, 2013.

- [51] Jalam R., “apprentissage automatique et catégorisation de textes multilingues”, thèse de doctorat, université lumière Lyon2, Juin 2003
- [52] Porter M., “An algorithm for suffix stripping”, pp 130-137, 1980.
- [53] Schmid H. “Probabilistic part-of-speech tagging using tree”, in conference on new methods in language processing, Manchester UK, 1994.
- [54] Rehel S., “ catégorisation automatique de textes et cooccurrence de mots provenant de documents non étiquetés ”, Mémoire présenté à la faculté des études supérieures de l’université laval, Québec, Janvier 2005.
- [55] Beekman M., Sword G.A., Simpson S.J., "Biological Foundations of Swarm Intelligence", chapitre Swarm Intelligence. Part of the series Natural Computing Serie, pp 3-41, 2008
- [56] Blum C. and Li X., "Swarm Intelligence in Optimization", chapitre Swarm Intelligence. Part of the series Natural Computing Serie, pp 43-85, 2008.
- [57] Abraham A., Guo H. and Liu H., " Swarm Intelligence: Foundations, Perspectives and Applications ", chapitre Swarm Intelligence Vol.26 of the serie Studies in computational Intelligence pp. 3-25, 2006.
- [58] El-Sayed H., Belal M., Almojel A. and Gaber J., "Swarm Intelligence”, Handbook of Bioinspired algorithms and application. 2005
- [59] Lim C.P. and Jain L., "Advances in Swarm Intelligence", innovations in swarm intelligence, pp. 1-7, Berlin, Germany, 2009.
- [60] Rennard J. P., "La Vie Artificielle ", 2003.
- [61] Abdou B., "Deux approches parallèles basées sur les colonies de fourmis pour la résolution du problème de la T- coloration des graphes ", mémoire d’ingénieur, ESI, Alger, 2009.
- [62] Baykasoğlu A., Özbakır L. and Tapkan P., "Artificial Bee Colony Algorithm and Its Application to Generalized Assignment Problems", swarm intelligence: focus on ant and particle swarm optimization I-Tech Education and publishing, Vienna, pp. 129-156, 2007.
- [63] Teodorović D., "Bee Colony Optimization (BCO)", chapitre Innovation in swarm intelligence Vol. 246 of the series Studies in Computational Intelligence pp.39- 60, 2009.
- [64] Lazinica A., "Particle Swarm Optimization", Published by In-Tech, 486 pp, 2009.

- [65] Martens D., Baesens B. and Fawcett T., " Editorial survey: swarm Intelligence for data mining ", Machine Learning, Vol. 82, Issue 1, pp 1-42, 2011.
- [66] Bonabeau E., Dorigo M. and Theraulaz G., " Swarm Intelligence From Natural to Artificial Systems ", Sante Fe Institute , Studies in the Sciences of Complexity, pp.38- 120, 1999.
- [67] Mullen R.J., Monekosso D., Barman S. and Remagnino P., "A review of ant algorithms", Expert Systems with Applications, Vol. 36, Issue 6, pp. 9608- 9617, 2009
- [68] Dréo J., Pétrowski A., Siarry P. and Taillard E., " Métaheuristiques pour l'optimisation difficile ", Editeurs Eyrolles, pp. 1- 368, 2003.
- [69] Dorigo M. and Krzysztof S., "An Introduction to Ant Colony Optimization ", Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle, IRIDIA Technical Report Series, pp. 1- 27, 2006
- [70] Costanzo A., Luong T.V. and Marill G., "Optimisation par colonies de fourmis ", pp. 1- 35, 2006.
- [71] Dorigo M., Birattari M. and Stutzle T., "Ant Colony Optimization", IEEE Computational Intelligence Magazine, Vol. 1, Issue 4, pp. 28- 39, 2006.
- [72] Deneubourg J.-L., Aron S., Goss S. and Pasteels J.-M., The self-organizing exploratory pattern of the Argentine ant, Journal of Insect Behavior, Vol 3, pp 159, 1990.
- [73] Dorigo M., Maniezzo V. and Colorni A., "Ant system: optimization by a colony of cooperating agents", IEEE Transactions on Systems, Man, and Cybernetics--Part B, Vol 26, N°1, pp. 29-41, 1996.
- [74] Dorigo M. and Gambardella L.M., "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem", IEEE Transactions on Evolutionary Computation, Vol. 1, N° 1, pp 53-66, 1997.
- [75] Stützle T. and Hoos H.H., "MAX MIN Ant System", Future Generation Computer Systems, Vol. 16, pp 889-914, 2000.
- [76] Handl J., "visualizing internet queries using ant-based heuristics", thesis, Monash University, Melbourne, Australia, Novembre 2001.
- [77] Ramos V. and Merelo J.J., "Self-Organized Stigmergic Document Maps: Environment as a Mechanism for Context Learning", In First Spanish Conference on Evolutionary and Bio-inspired Algorithms, Spain, pp. 284-293, 2002

- [78] Azaryuon K.. and Fakhar B., “A Novel Document Clustering Algorithm Based on Ant Colony Optimization Algorithm”, *Journal of mathematics and computer Science*, pp. 171-180, 2013.
- [79] Lumer E. D. and Faieta B., “Diversity and adaptation in populations of clustering ant”, in *Proceeding of the third international conference on simulation of adaptative behavior*, pp 501- 508, 1994.
- [80] Lachtar N. and Bahi H., “Application of an ant colony algorithm for text indexing ”, in *proceedings of the international conference on multi media computing and systems ICMCS’11* pp. 1-6, 2011.
- [81] Bali M.and Gore D. “Text Document Classification using Ant Colony Optimization and Genetic Algorithm”, in *International Journal of Innovative Research in Computer and Communication Engineering*, Vol.3, issue 12, pp. 1- 8 decembre 2015.
- [82] Solnon C., “Contributions à la Résolution Pratique de Problèmes Combinatoires des Fourmis et des Graphes”, *Mémoire pour l’Obtention de l’Habilitation à Diriger des Recherches*, Université Claude Bernard Lyon1, Décembre 2005.
- [83] Costanzo A., Luong T. V. and Marill G., “Optimisation par colonies de fourmis”, 2006
- [84] Berners-Lee T., “The Semantic Web”, *Scientific American Magazine*, 2001.
- [85] Plu J., “Introduction au Web sémantique”, disponible à [http://jp1u.developpez.com/tutoriels/web-semantique/introduction /](http://jp1u.developpez.com/tutoriels/web-semantique/introduction/) consulté le 01/12/2013.
- [86] Gruber T., “A Translation Approach To Portable Ontology Specifications, Knowledge Acquisition”, Vol. 5, issue 2, pp. 199-220, 1993.
- [87] Bouarab-Dahmani F., “Modélisation basée ontologies pour l’apprentissage interactif-application à l’évaluation des connaissances de l’apprenant”, these d’Université Mouloud Mammeri de Tizi Ouzou, Algérie, Novembre 2010.
- [88] Guarino N. and Giaretta P., “Ontologies and knowledge bases: Towards a Terminological Clarification”, in Mars N. J. I. (Ed.), *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, Amsterdam: IOS Press, pp. 25-32, 1995.
- [89] Kassel G., Abel M., Barry C., Boulitreau P., Irastorza C. and Perpette S., “Construction et exploitation d’une ontologie pour la gestion des connaissances d’une équipe de recherche”, in *actes de journées francophones d’ingénierie de connaissances*, 2000.
- [90] Bourdeau J. and Mizoguchi R, “Collaborative Ontological Engineering of Instructional Design Knowledge for an ITS Authoring Environment”,in *Intelligent Tutoring Systems* ,Springer, Heidelberg, pp.399-409, 2002.
- [91] Paquette G., “L’ingénierie pédagogique pour construire l’apprentissage en réseau”,

presses de l'Université du Québec, 2002.

[92] Uschold M., and King M., "Towards a methodology for building ontologies", in Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI'95, 1995.

[93] Gomez-Perez A., Fernandez M., and Vicente A. J., "Towards a Method to Conceptualize Domain Ontologies", in proceedings of the European Conference on Artificial Intelligence, ECAI'96. pp. 41-52, 1996.

[94] Kassel G., "OntoSpec: une methode de specification semi-informelle d'ontologies", in Actes des journées francophones d'Ingénierie des Connaissances", pp. 75-87, 2002.

[95] Fensel D., Horrocks I., Van Harmelen, F. and De S., "Oil in a nutshell", in Proceedings of European Knowledge Acquisition Workshop, Vol. 1937, pp. 1-16, Springer-Verlag, 2000.

[96] Hendler J. and McGuinness D., "The Darpa Agent Markup Language", <http://www.daml.org,2001>.

[97] Gruber T. and Olsen G., "An ontology for engineering mathematics", in Proceedings of the 4th International Conference on Principles of Knowledge Representation and Reasoning, 1994.

[98] Musique algérienne, (http://fr.wikipedia.org/wiki/Musique_alg%C3%A9rienne).

[99] Benbabaali S. and Rahal B., "La joie des âmes dans la splendeur des paradis andalous", ANEP, Alger (2010)

[100] Benbabaali S. and Rahal B., "La plume, la voix et le plectre", Poèmes et chants d'Andalousie, Alger, 2008.

[101] Boumedini B. and Dadoua-Hadia N., "Une analyse du discours raï algérien chez Cheb Hasni, Cheb Bilel et Chabba Kheira, Synergie Algérie N° 11, pp. 191-198. (2010)

[102] Boumedini, B., Dadoua-Hadia, N. : Le raï : Histoire d'une chanson née en Algérie", Revue des études afroeuropéenne 2010.

[103] Boumedini B. and Dadoua-Hadia N., "Catégorie d'emprunt dans la chanson Rap en Algérie L'exemple des Groupes : T.O.X, M.B.S et Double Canon", Synergie Algérie N° 8, pp. 139-147, 2009.

[104] Boumedini B. and Dadoua-Hadia N., "Paroles de jeunes à travers la chanson Rap en Algérie" Synergie Brizil N° 9, pp. 75-82. 2011

[105] Khan Latifur R., "Ontology-based Information Selection", Phd Thesis, Faculty of the Graduate School, University of Southern California. August 2000.

[106] Lesk M.E., “Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from a nice cream cone”, in Proceedings of the SIGDOC Conference, Toronto, 1986.