وزارةالتعليم العالي و البحث العلمي

BADJI MOKHTAR UNIVERSITY ANNABA

UNIVERSITE BADJI MOKHTAR ANNABA

جامعة باجي مختار – عنابة

Faculté des Sciences de l'Ingéniorat

Département d'Informatique

Thesis

Presented in view of obtaining the diploma of
Doctorate 3rd cycle

Entitled

Semi-Supervised Learning approach for Case-Based Reasoning systems: medical application

Domain: Mathematics and Computer Science

Field: Computer Science

Specialty: Computer Science

By

Chebli Asma

In front of the jury

Pr. Hayet Farida Merouani	University Badji Mokhtar-Annaba	**Reporter**
Dr.Akila Djebbar	University Badji Mokhtar-Annaba	**Co- Reporter**
Pr.Nabiha Azizi	University Badji Mokhtar-Annaba	**Chairwoman**
Pr.Smaine Mazouzi	University 20 Aout 1955-Skikda	**Reviewer**
Dr. Brahim Farou	University 08 Mai 1945-Guelma	**Reviewer**
Dr.Amine Khaldi	University Kasdi Merbah-Ouargla	**Reviewer**

Year 2020/2021

وزارةالتعليم العالي و البحث العلمي

BADJI MOKHTAR UNIVERSITY ANNABA

UNIVERSITE BADJI MOKHTAR ANNABA

جامعة باجي مختار – عنابة

Faculté des Sciences de l'Ingéniorat

Département d'Informatique

THÈSE

Présentée en vue de l'obtention du diplôme de
Doctorat 3ème cycle

Intitulé

Une approche semi-supervisée pour l'apprentissage des systèmes de raisonnement à partir de cas: application dans le domaine médical

Domaine: Mathématiques et Informatique

Filière: Informatique

Spécialité: Informatique

Par

Chebli Asma

DEVANT Le JURY

Pr. Hayet Farida Merouani	University Badji Mokhtar-Annaba	**Rapporteur**
Dr.Akila Djebbar	University Badji Mokhtar-Annaba	**Co- Rapporteur**
Pr.Nabiha Azizi	University Badji Mokhtar-Annaba	**Président**
Pr.Smaine Mazouzi	University 20 Aout 1955-Skikda	**Examinateur**
Dr. Brahim Farou	University 08 Mai 1945-Guelma	**Examinateur**
Dr.Amine Khaldi	University Kasdi Merbah-Ouargla	**Examinateur**

Année 2020/2021

To my dear parents with all my love and for everything they have given me.

To my sister and brother whom I love so much.

Not forgetting my little Bloom.

"On ne voit bien qu'avec le coeur. L'essentiel est invisible pour les yeux."

Le Petit Prince

Aknowledgments

First and foremost, I would like to praise and thank Allah, the almighty, who has granted countless blessing knowledge, and opportunities so that I have been finally able to accomplish the thesis.

Undertaking this PhD has been a truly life-changing experience for me and it would not have been possible to do without the support and guidance that I received from many people.

I would like to thank ***Pr.Hayet Farida Merouani***, Professor at the University of Badji Mokhtar Annaba, Algeria, who supervised me throughout this thesis and who shared her brilliant insights with me. I would also like to thank her for her kindness, her permanent availability and for the numerous encouragements she gave me. I would like to express my sincere gratitude to ***Dr.Akila Djebbar***, Senior Lecturer at the University of Badji Mokhtar, Annaba, Algeria, for her help in the preparation of this thesis. It was at her side that I understood what rigour and precision meant.

A special acknowledgement and my sincere gratitude to ***Pr.Hakim Lounis***, thanks to whom I had the chance to apply and be granted a scholarship at the Université du Québec à Montreal,Canada for which I will always be grateful. This scholarship really allowed me to carry my research and conduct my thesis, but much more than that, I came home after a year in Montreal with a lot of splendid memories to cherish forever.

I express my gratitude to ***Pr.Nabiha AZIZI*** for agreeing to chair the jury of this thesis. My thanks also go to ***Pr.Smaine MAZOUZI***, ***Dr.Brahim FAROU*** and ***Dr.Amine KHALDI*** for the honor of being my thesis jury members.

I would like to express my gratitude to my parents ***Souad and Farid*** to be so loving, invested and patient during all these years, but mostly for their support without which I could never have gotten so far in this journey.

A special gratitude to my sister ***Mina*** and brother ***Yassine***. ***Mina*** you were my companion during this PhD journey, I really believe with all my heart that this would have taken a different turn without you, I prefer by far the turn we have taken together. ***Yassine***, despite the 3 years that separate us I will never stop considering you as my little brother, thank you for being the brother you are, very kind and supportive. ***Bloom***, thank you sweet heart for all those nights spent by my side on the living room table, you are a balm to the soul. I know you didn't stay late at night just to get an extra bowl, or for me to give you treats without dad knowing, but because you love me.

To my grandmother, who constantly prays for us and I believe that it is through her prayers that I am able to overcome challenging times.

And of course my two best-friends (my safety blanket) ***Abir*** and ***Imen***, Without their tremendous understanding and encouragement these past few years, it would be impossible for to come so far.

Abstract

Case-based reasoning (CBR) concerns the study of intelligent decision systems based on past experiences. Strongly influenced by cognitive science, the initial concept of case-based reasoning evolved from results of several conducted studies concerning the human brain. The quality of these systems is directly related to the quality of their case bases, which makes the maintenance of the latter of great importance, as it can be performed at different stages of the case-based reasoning life cycle .

For the realization of this thesis, our attention has been drawn to the fact that most work on case-based reasoning systems focuses on the life cycle of the system once it is operational, or on the maintenance of knowledge container to avoid performance degradation after several reasoning cycles. However, *to exploit an implemented case-based reasoning system, it must first be developed*.

Little attention is paid to the development phase of case-based reasoning systems, or to the problems that may be encountered during their development. Considering the first two stages of development, namely data collection and acquisition of cases where labeled data are required, we can easily be confronted with the problem of collecting data which must then be processed, refined and structured into the form of cases (Problem, Solution). This makes the task of acquiring an initial case base difficult, as it is this that allows the system to be operational and to enable reasoning.

Given the scarcity of case bases, often assumed to exist or predefined by human experts, which is rarely the case , the first contribution consists of a preventive maintenance strategy at the development stage. Active learning is used in conjunction with semi-supervised learning to build and enrich the knowledge container with relevant and useful cases for reasoning.

Since Case-based reasoning systems are implemented to work over a long periods of time, this results in a rapid expansion of the case base, due to the retention of cases at the end of each life cycle. This can negatively affect the quality of the case-based reasoning outcomes and can slow the speed of the query execution time at the retrieval phase.

As such, we were interested in the second contribution to introduce a second maintenance strategy, in order to maintain or improve the quality of the case base built during the development phase, once the case-based reasoning system is operational. The objective is to reduce the size of the case base using a soft clustering technique namely, Fuzzy C-means to identify the relevant cases that should be saved and those that should be removed from the case base. The two proposed approaches have been validated on a number of databases and the results obtained are very encouraging.

Keys words: Machine Learning, Case-Based Reasoning, Case Base Maintenance, Semi-Supervised Learning , Active Learning, Sampling strategy, Clustering algorithm.

Résumé

Le raisonnement à partir de cas (RàPC) concerne l'étude des systèmes de décision intelligents basés sur des expériences passées. Fortement influencé par les sciences cognitives, le concept initial du raisonnement à partir de cas a évolué à partir des résultats de plusieurs études menées sur le cerveau humain. La qualité de ces systèmes est directement liée à la qualité de leur base de cas, ce qui rend la maintenance de cette dernière d'une grande importance, sachant qu'elle peut être effectuée à différentes étapes du cycle de vie du raisonnement à base de cas .

Pour la réalisation de cette thèse, notre attention a été attirée par le fait que la plupart des travaux sur les systèmes de raisonnement à base de cas se concentrent sur le cycle de vie du système une fois qu'il est opérationnel, ou sur la maintenance des conteneurs de connaissances pour éviter la dégradation des performances après plusieurs cycles de raisonnement. Cependant, *pour exploiter un système de raisonnement à base de cas, il faut d'abord le développer*.

Peu d'attention est accordée à la phase de développement des systèmes de raisonnement à base de cas, ou aux problèmes qui peuvent être rencontrés au cours de leur développement. Si l'on considère les deux premières étapes du développement, à savoir la collecte de données et l'acquisition de cas où des données étiquetées sont nécessaires, on peut facilement être confronté au problème de la collecte de données qui doivent ensuite être traitées, raffinées et structurées sous forme de cas (Problème, Solution). Ceci rend difficile la tâche de disposer d'une base de cas initiale, car c'est elle qui permet au système d'être opérationnel et de rendre le raisonnement possible.

Étant donné la rareté des bases de cas, souvent supposées exister ou prédéfinies par des experts humains, ce qui est rarement le cas, la première contribution consiste en une stratégie de maintenance préventive au stade du développement. L'apprentissage actif est utilisé en conjonction avec l'apprentissage semi-supervisé pour construire et enrichir le conteneur de connaissances avec des cas pertinents et utiles pour le raisonnement.

Comme les systèmes de raisonnement à base de cas sont mis en œuvre pour fonctionner sur de longues périodes, il en résulte une expansion rapide de la base de cas, en raison de la mémorisation des cas à la fin de chaque cycle de vie. Cela peut affecter négativement la qualité des résultats du raisonnement à base de cas et ralentir la vitesse d'exécution des requêtes lors de la phase de remémoration.

Ainsi, nous nous sommes intéressés dans la deuxième contribution à introduire une deuxième stratégie de maintenance, afin de maintenir ou d'améliorer la qualité de la base de cas construite pendant la phase de développement, une fois que le système de raisonnement à base de cas est opérationnel. L'objectif est de réduire la taille de la base de cas en utilisant une technique de partitionnement souple, à savoir le partitionnement flou, afin d'identifier les cas pertinents qui doivent être sauvegardés et ceux qui doivent être supprimés de la base de cas. Les deux approches proposées ont été validées sur un certain nombre de bases de données et les résultats obtenus sont très encourageants.

Mots clés: Apprentissage automatique, Raisonnement à Partir de Cas, Maintenance base de cas, Apprentissage Semi-Supervisé, Apprentissage Actif, Stratégie de selection, Algorithme de partitionnement.

الملخص

يدور المنطق القائم على الحالة حول دراسة أنظمة القرار الذكية بناءً على التجارب السابقة. متأثرًا بشدة بالعلوم المعرفية ، تطور المفهوم الأولي للتفكير القائم على الحالة من نتائج العديد من الدراسات التي أجريت على الدماغ البشري. ترتبط جودة هذه الأنظمة ارتباطًا مباشرًا بجودة قاعدة الحالة الخاصة بها ، مما يجعل الحفاظ على الأخير ذا أهمية كبيرة ، مع العلم أن هذه الصيانة يمكن إجراؤها في مراحل مختلفة من دورة حياة التفكير بناءً على الحالة .

لتحقيق هذه الأطروحة ، تم لفت انتباهنا إلى حقيقة أن معظم الأعمال المتعلقة بأنظمة التفكير القائم على الحالة تركز على دورة حياة النظام بمجرد تشغيله ، أو على صيانة حاويات المعرفة لتجنب تدهور الأداء بعد دورات التفكير المتعددة. ومع ذلك ، لاستغلال نظام التفكير القائم على الحالة ، يجب أولاً تطويره.

أنظمة التفكير القائم على الحالة ، أو المشاكل التي قد تواجه أثناء تطويرها. إذا نظر المرء في المرحلتين الأوليين من التطوير ، وهما جمع البيانات والحصول على الحالات التي تتطلب البيانات المصنفة ، فيمكن بسهولة مواجهة مشكلة جمع البيانات التي تحتاج بعد ذلك إلى المعالجة والتنقيح والتنظيم في شكل حالات (مشكلة،حل). هذا يجعل مهمة الحصول على قاعدة حالة أولية صعبة ، لأن هذا هو الذي يسمح للنظام بالعمل وجعل التفكير ممكنًا.

نظرًا لندرة قواعد الحالة ، التي غالبًا ما يُفترض وجودها أو تم تحديدها مسبقًا من قبل خبراء بشريين ، وهو ما نادرًا ما يحدث ، تتكون المساهمة الأولى من استراتيجية الصيانة الوقائية في مرحلة التطوير. يتم استخدام التعلم النشط جنبًا إلى جنب مع التعلم شبه الخاضع للإشراف لبناء وإثراء حاوية المعرفة بالحالات ذات الصلة والمفيدة للاستدلال.

نظرًا لأن أنظمة الاستدلال المستندة إلى الحالة يتم تنفيذها للعمل على مدى فترات زمنية طويلة ، فإن هذا يؤدي إلى التوسع السريع في قاعدة الحالة ، بسبب الاحتفاظ بالحالات في نهاية كل دورة حياة. يمكن أن يؤثر ذلك سلبًا على جودة نتائج الاستدلال المبني على الحالة ويبطئ سرعة تنفيذ الاستعلام أثناء مرحلة الاستدعاء.

وبالتالي ، في المساهمة الثانية ، كنا مهتمين بتقديم إستراتيجية صيانة ثانية ، من أجل الحفاظ على جودة قاعدة الحالة التي تم إنشاؤها أثناء مرحلة التطوير أو تحسينها ، بمجرد تشغيل نظام التفكير القائم على الحالة. الهدف هو تقليل حجم قاعدة الحالة باستخدام تقنية تقسيم مرنة ، أي التقسيم الغامض ، لتحديد الحالات ذات الصلة التي يجب حفظها وتلك التي يجب حذفها من قاعدة الحالة. تم التحقق من صحة النهجين المقترحين في عدد من قواعد البيانات والنتائج التي تم الحصول عليها مشجعة للغاية.

لكلمات المفتاحية:

تعلم الآلة ، التفكير القائم على الحالة ، الصيانة القائمة على الحالة ، التعلم شبه الخاضع للإشراف ، التعلم النشط ، إستراتيجية الاختيار ، خوارزمية التقسيم.

Contents

List of Figures

List of Tables

List of acronyms

AI	**A**rtificial **I**ntelligence
ASSM	**A**ctive **S**emi **S**upervised **M**aintenance
AL	**A**ctive **L**earning
CAD	**C**omputer **A**ided**D**iagnosis
CB	**C**ase **B**ase
CBM	**C**ase **B**ase **M**aintenance
CBR	**C**ase **B**ased **R**easoning
C IRD	**C**lustering _ **I**nformative **R**epresentative **D**ivers
DAG	**D**irect **A**cylic **G**raph
FCM	**F**uzzy **C** **M**eans
MCS	**M**ulti **C**lissifier**S**ytem
LMT	**L**ogistic **M**odel **T**ree
ML	**M**achine **L**earning
PCC	**Pe**rcentage **C**orrect **C**lassification
S	**S**ize
SSL	**S**emi **S**upervised **L**earning
SMO	**S**equential **M**inimal **o**ptimization
SVM	**S**upport **V**ector **M**achine

S3VM **S**emi **S**upervised **S**upport **V**ector **M**achine

TSVM **T**ransductive **S**upport **V**ector **M**achine

General introduction

Contents

0.1 Background

Influenced significantly by the cognitive sciences, the early concept of Case-Based Reasoning(CBR) evolved from the results of several studies conducted on the human brain. Under different definitions in the literature, CBR is considered as: Reasoning by remembering, Reasoning for reminding, an approach to problem solving and learning, and it is defined as a sub field of artificial intelligence. Among the diverse tracks of artificial intelligence, case-based reasoning mimics the human reasoning process. It is therefore a methodology that has shown great promise in various domains for the fulfilment of several tasks. Significant work in the field of medicine using the CBR approach has been notable for several decades now [19],[20], particularly for the implementation of computer-aided diagnosis (CAD) systems, which allow physicians to be guided in real time to make a diagnosis.

It is worth noting that several machine learning methods,including CBR, have been widely and successfully used in the implementation of computer-aided diagnosis (CAD) systems, in an attempt to boost the diagnostic capacity of physicians and reduce the time needed for an efficient diagnosis. However, these methods rely on considerable volumes of diagnosed (supervised/labeled) instances required to achieve a certain efficiency. They consider hypothesis derived from a large amount of pre-diagnosed samples (medical data), i.e. data collected from a number of medical examinations actually performed and their respective diagnoses made by medical experts.

Nevertheless, in practice, unlabeled data is often the most abundant and offers a great richness of information, but the task of labelling this data is considered to be a burden for human experts, as it is a time consuming and a costly process that often requires the intervention of an expert. In this regard, semi-supervised learning (SSL) integrates unlabeled data into the prediction model. In this sense, semi-supervised learning is halfway between supervised and unsupervised learning: SSL seeks to exploit unlabeled data to learn the relationship between examples and their labels.

0.2 Problem statement

The implementation of an efficient CBR system is explicitly associated to the quality of the case base (CB), an essential knowledge container. CBR is a memory-oriented cognitive model that focuses on how to acquire new skills or generate hypotheses for new situations on the basis of previous experiences. This artificial intelligence approach depends heavily on the performance of the case base to make highly adequate decisions. The quality of the latter is particularly crucial when considering the implementation of a CAD system using a CBR framework. CBR is particularly well suited in medecine, as it memics the experts reasoning process: medical experts use the knowledge they have gained from books and ex-

periences in exactly the same way that CBR works: by learning by remembering cases.

Therefor, case base maintenance becomes of great importance when one is interested in the implementation of a computer-aided diagnosis system using the case-based reasoning (CBR) approach. This draws our attention to the fact that most work on case-based reasoning systems focuses on the life cycle of the system once it is operational, or on the maintenance of knowledge containers to avoid performance degradation after several reasoning cycles. However, *to exploit an implemented case-based reasoning system, it must first be developed*.

Little attention is paid to the development phase of case-based reasoning systems, or to the problems that may be encountered during their development. Considering the first two stages of development, namely data collection and acquisition of cases where labeled data are required, we can easily be confronted with the problem of collecting data which must then be processed, refined and structured into the form of cases (Problem, Solution). This makes the task of having an initial case base difficult, as it is this that allows the system to be operational and to enable reasoning.

Case-based reasoning is a method whose name alone explains its purpose well enough. The core of these systems is the case base, a primary container of knowledge that allows the CBR cycle to proceed and the reasoning to be conducted, in order to deliver adequate solutions to the problems that are presented to the systems. CBR is an artificial intelligence method that mimics the human reasoning process, and within this framework the case base represents the human brain.

The question that emerges is: «Can a brain with little experience or knowledge cope with a large number of diverse problems? and solve them efficiently? having an adequate solution to each presented problem?». The answer is most definitely NO, and the same goes for a limited and non-diverse case base.

0.2.1 Thesis Statement

According to Creswell [21] researchers usually write at the very least a main research question and sub-questions. This thesis raises and answers a series of research questions, this section summarises and formulates these questions in the form of emerging aspects. Based on the issues discussed earlier, four main aspects emerge and are addressed, in order to build a quality case base and to cope with the lack of a labeled case bases:

1. Build a quality case base at the development stage of the CBR system given a small set of labeled data, and monitor the storage size of the CB to avoid retention of irrelevant cases,
2. In order to cope with the scarcity of labeled case bases necessary for the implementation of a reliable CBR system for computer-aided diagnosis, the consideration of unlabeled data through semi-supervised learning is required,
3. The impact of active learning in the semi-supervised framework on the performance of the classification module. Where active learning seeks the minimisation of the labeling cost of unlabeled data points deemed to be valuable for learning,
4. Maintain the quality of the case base acquired during the development phase once the CBR system is operational. Given the risk of degradation of the quality of the case base after several reasoning cycles that lead to a rapid growth of the case base.

0.3 Purpose of the study and Contributions

The main objective of this work is to investigate an approach to build and maintain a case base during the early stages of the implementation of a CBR system, namely at the development phase.

The concept of «**preventive action**» exists in several fields of application, the initial purpose of a preventive action is to eliminate a perceived weakness within the system or the origin of a suspected undesirable situation in order to prevent it from occurring i.e. it corrects problems before they happen. A preventive action also aims to improve the efficiency of the system.

The first proposed approach can be perceived as a preventive action, which qualifies it as a preventive maintenance strategy. In engineering preventive maintenance is a preventive action which aims to reduce the probability of failure of an element(in our study the case base) or the degradation of the performance of a service provided (in our case the service is reasoning), in order to prevent an equipment from failure (the CBR system can be perceived as an equipment) [22].

We are mindful of the problems that can arise using a limited case base, instead of launching a CBR system with a small case base, which may very likely lead to many reasoning mistakes and a poorly performing system, we address these potential problems before they occur, and prevent them from arising once the system is operational. Indeed, what allows us to qualify our approach as a preventive maintenance strategy is that it shares exactly the same objectives as this type of strategy:

1. Increase the reliability of the system,
2. Improve the availability of the system,
3. Reduce failure cost.

Our maintenance strategy uses machine learning techniques to enrich the knowledge container known as the case base with relevant and useful data for reasoning (maintaining the case base while building it, by storing only valuable cases). The first contribution consists of a maintenance strategy that aims at taking advantage of unlabelled data, using active learning in conjunction with semi-supervised learning to select the instances judged as valuable from the pool of unlabeled data points. The goal is to use this pool of unlabeled data along with the few labeled data available in order to build a quality case base given the scarcity of such bases supposed to exist or predefined by human experts, which is not always the case.

A second maintenance strategy is proposed, in order to maintain the quality of the case base built during the development phase, once the CBR system is operational. Given the risk of degradation of the quality of the case base after several reasoning cycles that lead to a rapid growth of the case base. The case-based reasoning system is built to run for long periods of time, adding cases to the case base through the retention phase. As a result, the case base can grow very quickly, which can negatively affect the quality of the CBR outcomes and slow the speed of the query execution time at the retrieval phase. To ensure the continuous quality of the system, we propose a second maintenance strategy.

The objective is to reduce the size of the case base using a clustering technique, to identify the relevant cases that should be saved and those that should be removed from the case base to maintain or improve the quality of CBR as much as possible.

0.4 Outline of the Thesis

This thesis is organised into five chapters, besides the chapter of the general introduction. The first three describes the theoretical background and foundations of the approaches proposed in this thesis. The last two chapters describe the main contributions of this thesis (Figureo.1). It is organised as follows:

The first chapter (**Case-Based Reasoning**), we have presented the research axis that inspired our research topic, namely CBR. We discussed the essential points concerning this methodology, starting with the fundamentals necessary to familiarize with the different terms of the approach. Thereafter, the structuring and representation of the case were presented, along with the case indexing and the case base organization. Afterwards, the CBR life cycle was discussed along with its detailed steps that allow the manipulation of this knowledge mechanism. Finally, we concluded this chapter with some CBR applications, discussing how CBR can be used, and the characteristics that allow this approach to be employed in certain domains.

The second chapter is (**Case Base Maintenance**), we further explored the field of case base maintenance. Understanding the nature of maintenance process and how it is related to the overall CBR process is advantageous for identifying good research opportunities and appreciating maintenance practice. This allowed us to understand the different courses on which maintenance can be performed, which draw attention to the fact that most existing CBR papers focusing on the life cycle of the system once it is operational, but rarely on the stage of development or the problems that can be encountered when trying to develop a case based reasoner, namely, the acquisition of data to build a CB for CBR, as it is a crucial step for the development of the system, as this knowledge container represents the heart of CBR.

The third chapter (**Machine Learning techniques**) introduced machine learning techniques, with a particular focus on on the semi-supervised and active learning used for the implementation of our proposed approach.

Chapter four (**Maintenance at the development stage: Active Semi-Supervised Maintenance(ASSM)approach**) describes our first contribution which is an Active Semi-Supervised Maintenance (ASSM) strategy to build and maintain a case base at the early stages of development of CBR systems. Performance criterion was considered to evaluate the quality of the case bases. The evaluation of the proposal demonstrates the effectiveness of ASSM which is interesting as a CBM strategy, able to be efficient in terms of a controlled growth of the storage size and scoring satisfying classification.

The last chapter ,chapter 5 (**Clustering Informative, Representative and Divers cases (C_IRD)** describes our second contribution, where we seek to maintain or even improve the quality of the case base built during the development phase (first contribution) which might have degraded after several CBR reasoning cycles. The contribution presented in Chpater 5 is a maintenance approach that addresses the draw-backs that follow an operational CBR system and the blind retention of cases.

The conclusion highlights the advantages of the proposed approaches, while discussing the results obtained throughout this thesis and the perspectives to be given to this work.

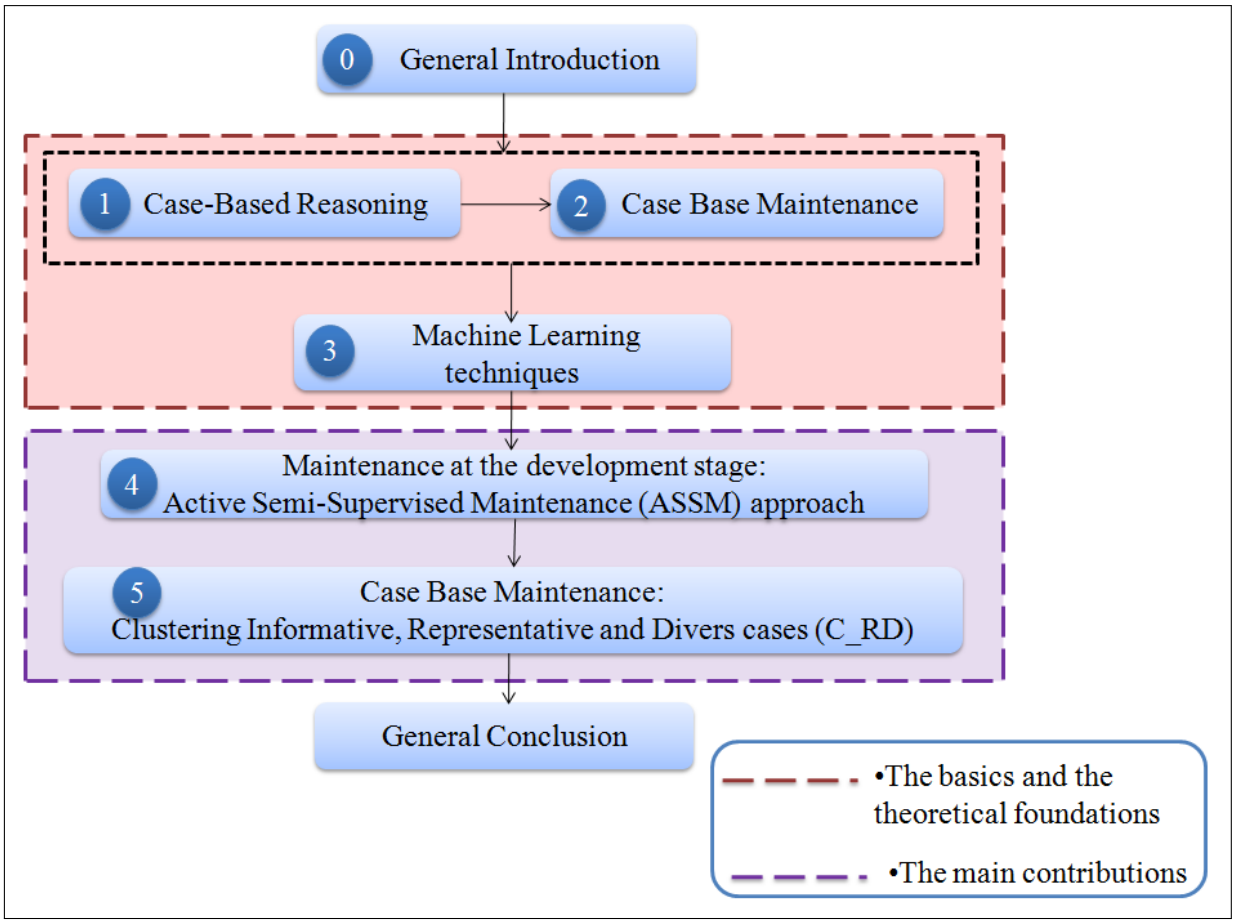

Figure 0.1 – *Thesis Map*

.

Case-Based Reasoning (CBR)

Contents

1.1 Introduction

Would you go under the knife for a brain surgery and trust a surgeon with little or no experience in the field? To solve complex problems, experience is mandatory. Experience is gained over time, as we go through life, this experience is utilized to help us deal with situations that are encountered on a daily basis. For instance, a doctor employs his experience gained from previous patients and treatments that worked with them, in order to treat a patient with similar symptoms. Same for a mechanic, his experience gained from previous engine problems is used to fixed new ones.

Case-based reasoning (CBR) concerns the study of intelligent decision systems based on past experiences. Strongly influenced by cognitive science, the initial concept of CBR evolved from results of several conducted studies concerning the human brain [23]. Published research papers about CBR in different journals demonstrates the extent of importance given by researchers to this methodology.

CBR is a branch of Artificial Intelligence (AI) that brings together machine learning and reasoning techniques in order to solve new problems by adapting solutions that worked for similar past experiences, these past experiences are stored as cases, to form a case base (CB) [5]. Unlike other problem-solving approaches in AI, CBR mimics the human way of thinking, it is memory based, which makes it very similar to the human reasoning process.

For over three decades now, CBR has been a flourishing field, and that is due to numerous reasons we will be explaining in this chapter, as it is the main research axis of our thesis. In section2 we present the fundamentals of CBR systems, necessary for understanding this AI approach, which will allow us on one hand to familiarize ourselves with the different terms we will be employing throughout the rest of this thesis.

CBR methodology allows the manipulation of knowledge in order to solving problems by finding similar situation modeled in the CB, and adapting the previous similar situations to the one in consideration. The knowledge modeling requires filed expertise, which is represented in form of cases. In section 3 we discuss the structure and representation of the case, which has implications on the type of the CBR model manipulated. We present the different models used in literature, as well as the most used model, along with CB organization and case indexing. The cases are stored in a memory called the "case base" (CB), considered as the center of any CBR life cycle. This life cycle implements reasoning by analogy and has several phases(steps) whose role is to manipulate the system knowledge in order to achieve the objectives set, as problem-solving tasks and/or knowledge acquisition. Indeed, CBR does not only support reasoning, it associates problem-solving with continuous learning[24], as it relies on experimental knowledge under the form of previous problem/solution patterns. In section 4 we discuss the different proposed representation of the CBR life cycle, the application and maintenance phases, and a discussion of each step that constitute the CBR life cycle. Finally, in Section 5, we have presented different applications of CBR in many areas and the tasks that can be performed, and have presented characteristics to distinguish whether the CBR approach would be applicable to certain areas or not.

1.2 Fundamentals of CBR

Strongly influenced by cognitive science, the initial concept of CBR evolved from results of several conducted studies concerning the human brain[23]. CBR is found under different definitions in literature; it is seen as: Reasoning by remembering[25], Reasoning for reminding[26], an approach to problem solving and learning[27], and it is defined as a sub field of artificial intelligence by Bergmann et al.[28].

CBR was originally inspired by the work of Minskey and Schank in the late 1970's[1]. Shank [29] formulated for the first time the paradigm of Case-Based Reasoning , but it was only until the end of the 1980s that the research in the field of CBR really began to take shape. Particularly with the DARPA conferences in the USA in 1988[30], before making its mark in Europe with the first European conference in 1993 at Kaiserslautern [31]), and again with the first international conference in Lisbon in 1995 [32].

1.2.1 Communities in CBR

Case-Based Reasoning is a methodology with roots in artificial intelligence(AI), cognitive science and knowledge engineering (Figure1.1). CBR means solving problems based on previous experiences, remembering past situations/cases to guide the solution of a new problem. It is a study that concerns intelligent decisions based on past experiences, a field in which memory models are studied and categorized.

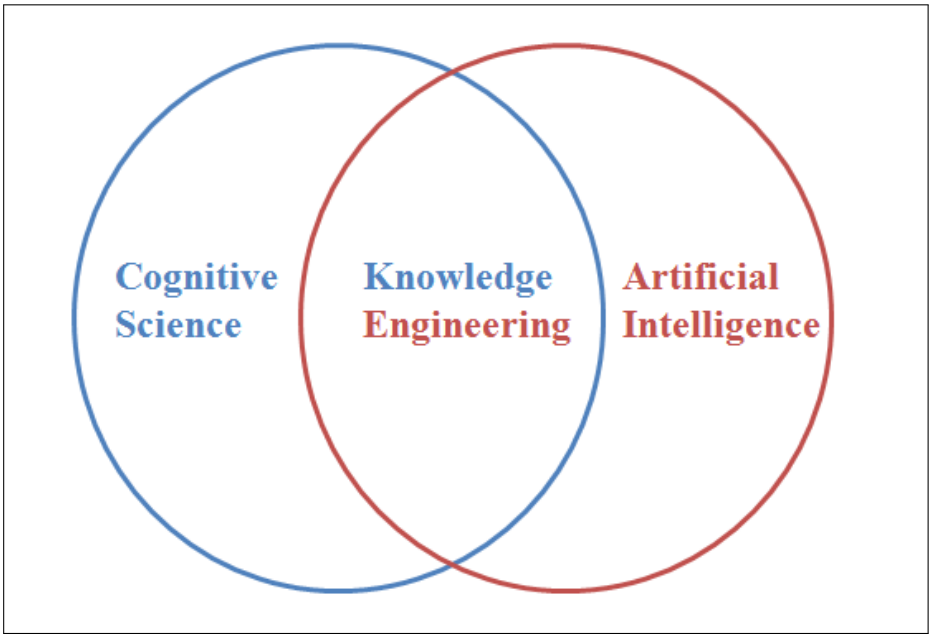

Figure 1.1 – *Communities of CBR (knowledge engineering at the intersection of AI and cognitive science)[1].*

1. **Artificial intelligence(AI)** it is a research field of science and engineering for developing intelligent systems. AI searches for ways to endow computer programs with intellectual capacities comparable to those of human being[33],
2. **Cognitive science** it can be defined as the scientific study of minds and brains and their processes, be they real, animal or artificial thought, and more generally of any cognitive system[34],
3. **Knowledge engineering**is the field that corresponds to the study of concepts, methods and techniques that allow the developing of knowledge based systems in any application domain, in order to help humans to carry out tasks with little or no prior formalization[35].

In brief, given a case to solve, case-based reasoning includes the following steps[36]:

- Retrieve relevant cases from the CB (an appropriate indexing of the CB is required),
- Select a set of best cases;
- Deduct a solution,
- Evaluate the solution (to assure that poor solutions are not repeated),
- Store the new solved case in the CB.

CBR is used complete a range of reasoning tasks, such as classification, planning and design.However, the key to the development of a successful CBR system is to limit its scopes to a single reasoning task. Known as a lazy learning method, CBR system can be built without the necessity to learn data specifics or patterns, just by taking the data coming from a data base.

Yet, CBR demonstrated to be very useful in many real world application domains. Some of the reasons are[4]:

- CBR falls under the intersection of numerous disciplines as mentioned earlier, which open the door to its adoption for diverse applications;
- CBR methodology mimics the human reasoning process. Therefore, when implementing a CBR system, we are using a human paradigm in a computational framework; while benefiting from the large memory and speed supplied by a computer;
- CBR does not require a complexes formalization of the problem and is able to deal with informal questions .

1.3 The case base

The case base is one of the four sources of knowledge required in CBR(Figure1.2)[4]. The four knowledge containers are: **The vocabulary** a container dedicated to the description of problems and solutions in the domain. **The similarity measure** encompasses knowledge about cases, and how they are compared to each other. **The case base (CB)** the core of the CBR system, as it contains the set of previously solved problems. And finally, **Adaptation knowledge** it defines how a retrieved solution is adapted to correspond to a new problem. This combined knowledge is employed to complete the CBR process. In the next section we deal with the case base and the cases stored in it.

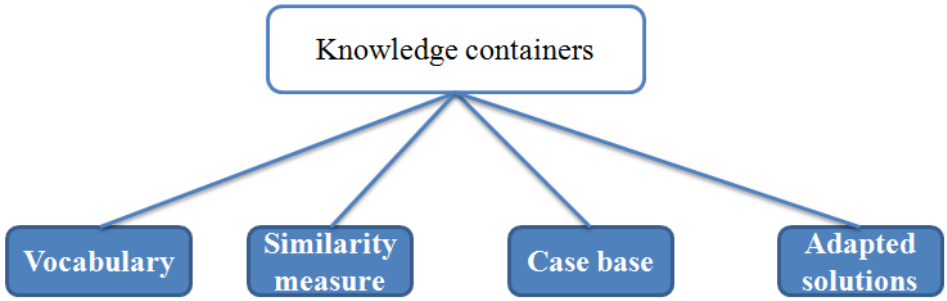

Figure 1.2 – *Knowledge containers(based on [2]).*

1.3.1 Case structuring

The idea of a "case" is to capture information as used in cognitive science to be used for problem-solving[37]. A case is an instance of a problem-solving process, in CBR it is generally composed of two disjointed spaces, they are the two component of the case that need to be distinguished: the problem description and the solution. The former embody the goals, constraints, initial data and task description. The last part comprehends the solution as it is, the steps to attain the solution (or trace), justification and annotation of the solution, along with alternative solutions and expectations (what's expected to happen when obtaining the solution) [3].

Bergmann, Kolodner and Plaza [37] described the following elements of case structure:

- A situation and its goal;
- The solution and, sometimes, the means to obtain it;
- The result of its execution;
- Explanation of results;
- Lessons that can be learned from the experience.

Furthermore, in the case representation, the outcome of a solution can be captured, if the latter has achieved the desired outcome or not. Accordingly, a case can be represented by the following tuple <p,s,o>, where p is the problem, s the solution and o the outcome. Yet, this is not an exhaustive description of the possible case components, other components can be considered.

Two types of cases can be distinguished: *source case* and *target case*[1]. The source case is the one in which the «problem »and «solution «parts are available. Thus, this case can be used to solve new problems. As for the target case, it is the one that carries the problem and whose solution is not available. Depending on the nature of the problem dealt with, there are several case representations. Traditional approaches classify them into three categories:

1. Textual representation,
2. Semi-structured representation (component vector),
3. Structured representation.

However, structured case representation is the most widely used in the majority of the works. Thus, the case is often represented as a set of descriptors.

A descriptor d is defined by a pair *d=(a,v)*, where*«a»* is an attribute and *«v »* is its associated value. A *source case* is represented by a pair (srce,Sol(srce)), and the *target case* by the pair (target,Sol(target)), where Sol(target) is unknown and for which we would like to provide a result. As the cases are represented by a set of descriptors then[1] :

- ds_i (for i=1,..,n): represents the descriptors of the problem part of the source case «srce »;
- dc_i (for i=1,..,n): represents the descriptors of the problem part of the target case «target»;

- Ds_i (for i=1,..,n): represents the descriptors of the solution part of the source case «Sol(srce)»;
- Dc_i (for i=1,..,n): represents the descriptors of the solution part of the target case «Sol(target)».

1.3.1.1 *Problem description*

CBR is problem-centered, as it is the principal purpose of the methodology: problem-solving. The formulation of a problem is related to the context in which it is stated, hence, each problem formulation requires a different type of solution. For example: *What is the price of this phone?*

1. One answer could be: To expensive for us;
2. Another answer could be: 1400$.

In order to find the suitable answer, it is important to know the context in which the problem is stated. For an accurate statement, the context need to be included in the problem formulation [4]. In the framework of CBR methodology, we refer to two types of problems: The problems in the CB, registered as experiences. These cases are candidate cases for reuse. However, the CBR process is triggered by a problem, a new problem that motivated the user to look for a problem-solving method. This problem is referred to as query problem, or simply, the problem. Essentially, the commonly used terms are: query instead of problem, and answer instead of solution.

As discussed in the previous section, ***Attribute-Value*** pairs is the simplest and commonly used representation. A sequence of features is used to describe a problem *(f1,...,fn)*.

Let us take as an example corona virus (Covid-19) prognosis. Considering a CBR system in charge of identifying whether a person is having Covid-19 or not. For that, some information about the patient should be gathered as cases. A possible attribute representation for such scenario is presented in Table1.1. The solution is the **Infected** attribute.

Table 1.1 – *Attribute-Value pair representation for COVID-19 example.*

Attribute	**Value**
Age	61
Sex	Female
High	1,65 m
Weight	60 kg
Fever	98.115° F
Body pain	Yes
Runny nose	Yes
Difficulty breathing	No
Infected	Yes

Another representation that could be used is ***Object representation*** , as it is very difficult to handle hundred of feature, grouping them by category can simplify the task. Object representation for the previous Covid-19 example is provided in Table 1.2 . This representation is not often used because, from a practical point of view, it can be reduced to a representation by attribute pairs.

The third main type of case representation is Relationship objects, commonly visualized as a tree or graph. For this type representation, the is no homogeneous way to represent all cases, the attributes cannot be localized by their position. To identify the attribute from the root of the graph a needed use of attribute name along with the path is required. For this knowledge representation scheme we take the Breast cancer prognosis, as shown in Figure1.3 [3].

Table 1.2 – *Object representation for COVID-19 example.*

Epidemiological data	Sex Female Age 61 Hight 1.65 m Weigh 60 Kg
Personal data	Fever 98.115° F Body pain Yes Runny noise Yes Difficulty breathing No Infected Yes

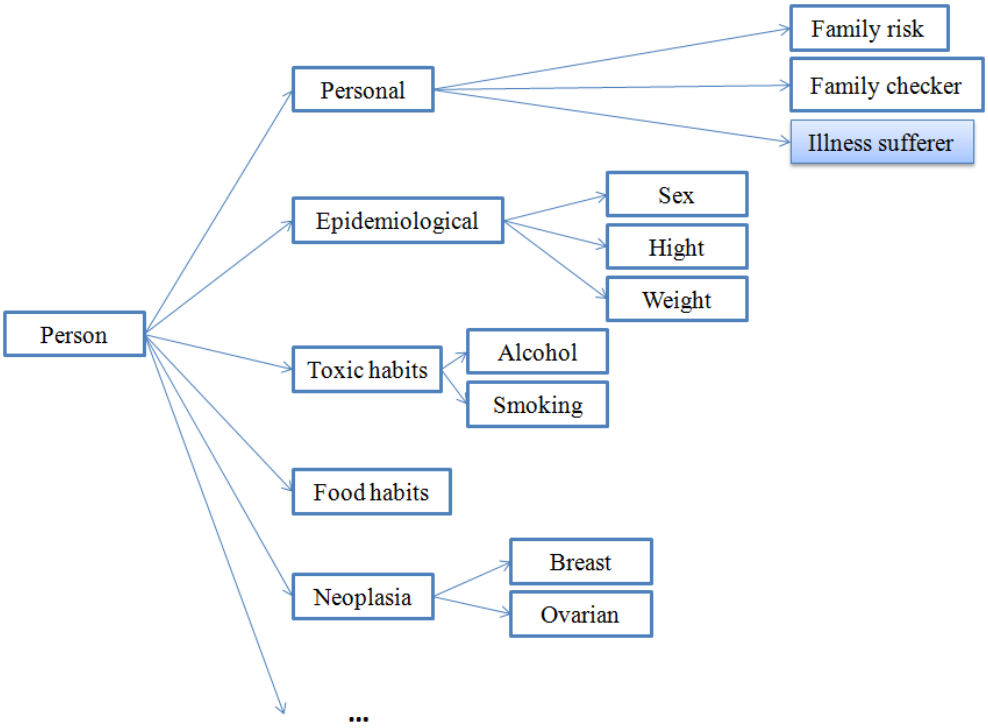

Figure 1.3 – *Relational representation of cases for the breast cancer example [3].*

1.3.1.2 *Solution types*

The information about the solution of a case depends on the problem-solving task[3]. Frequently, CBR is used to predict a label of a class (classification), given a set of labels L, the solution of the case $s \in L$. Binary classification is widely used, where two labels indicate a positive or a negative outcome. For instance, predicting if a device will fail or not in the near future, or given information about a person, if the later will suffer from an illness or not (prognosis). Yet, some domains require more than two labels, as an example credit approval domain, where we can have: low, medium and high risk of credit approving for an individual. This is known as multi-class labeling, it involves assigning to the solution of a problem a subset of labels $s \subseteq L$. Another example of multi-class labeling is diagnosis. A solution can be represented in a variety of ways[4]:

1. Can be just a solution in the narrow sense;
2. The solution can include :
    - Remarks on the strategy used to obtain the solution
    - Constraints restricting the solution's application
    - alternative solutions
    - ...

1.3.1.3 *Outcome*

When outlining the case structure, one of the design decisions is to represent and store information about how the solution solves the problem[3]. This implies acknowledging the fact that there will be cases with incorrect solutions in the CB which can be taken advantage of to avoid repetition. Only few CBR systems include such information, because a failure is handled at the retain stage, where learning methods are used to avoid future failure.

1.3.2 Case indexing

In the context of CBR methodology, a CB is a collection of cases used for the purpose of problem solving tasks. A CB is defined as : «A collection of structured set of cases »(adapted from [4]). A CB is generally a finite source of data, the particular point concerning case-based reasoning, is how the case base is used. The usage of CBR demands special ways to handle the CB, heavily referred to as " memory" in cognitive science. For this purpose, the CB should be organized into a manageable structure for an effective search and retrieval. Especially when we have a large memory, a simple linear organization for instance a list, is very inefficient for retrieval[36].

Case base organization relates to how cases are indexed and retrieved from memory. Pure CBR lazy approaches do not employ any indexing mechanism [3]. However, in order to facilitate the CB organization, and thus search for the most appropriate case(s) for the problem at hand it is necessary to index the cases.

It should be noted that when searching for appropriate cases, it is the problem part that will be associated. The problem part is described by a set of relevant characteristics called «indexes». Case indexing requires assigning indexes to cases to facilitate their retrieval, numerous guidelines on indexing have been proposed in the CBR context [38]. The index determines the context and situation in which the case will be searched and retrieved to be proposed for a given problem. It is necessary to find a way to manipulate the indexes for an optimal configuration.

Both manual and automated methods have been employed to select indexes. In the case of *manual indexing*, it is assumed that the purpose of using the case, including the circumstances under which the case will be useful, is accurately described. Yet, *automated indexing* are increasingly used. Moreover, the choice of indexes depends on the field of application, As Kolodner [39] points out , these indexes must verify some of the following properties, and index should be:

- *Predictive enough* in order to play a determining role in the choice of a solution for a new problem,
- *Abstract enough* to allow the case to be used several times for the resolution of several problems,
- *Concrete enough* for the case to be recognized as quickly as possible for the resolution of new problem.

1.3.3 Case base organization

Systems using CBR as a problem-solving approach may have several models. Case base organization is considered once the indexes have been chosen. We have three main types of case organization[4]: Flat, structured and semi-structured. Figure1.4 illustrates the three fundamental types of organization.

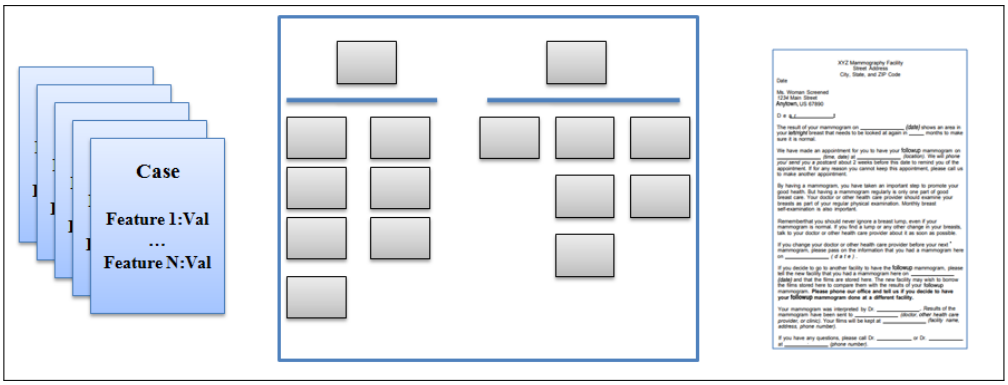

Figure 1.4 – *Main types of case organization: flat, structured, semi-structured [4]*

1. Flat organization: a widely used case base organization in CBR works, as it is simple to implement and suitable to manage small cases. Cases are organized within a table, where rows represents cases and columns are attributes.

2. Structured organization: cases can be organized into a structure: hierarchies and networks. Structured organization is related to the relational case representation, where cases are stored according to the relations of attributes.

3. Semi-structured organization: for this type of organization there is no specific given schema to represent cases in a uniform way within the case base, for this particular reason they are called loosely structured or unstructured. For this type the cases are usually hidden within texts or images[4].

We conclude that a case may therefore have several representations. In addition , a case in the field of medical diagnosis/prognosis has a very specific formalization this will be the subject of the next subsection.

-Example of a case suitable for medical diagnosis/ prognosis:

In terms of medical diagnosis/prognosis, the case requires an adapted structure, it is characterized by symptoms or medical features and their values. The objective is to assign to each case a label that is the appropriate diagnosis/prognosis taken into account the symptoms or medical features. The structure of cases could be as follow[40]:

$$Problem \Longleftrightarrow Symptoms/medical\,features$$

$$Solution \Longleftrightarrow Diagnosis/prognosis$$

An example of the structure of a case dedicated to diagnosis is demonstrated in Figure1.5, the representation used in this example is a vector type representation (list of attribute-value pair).

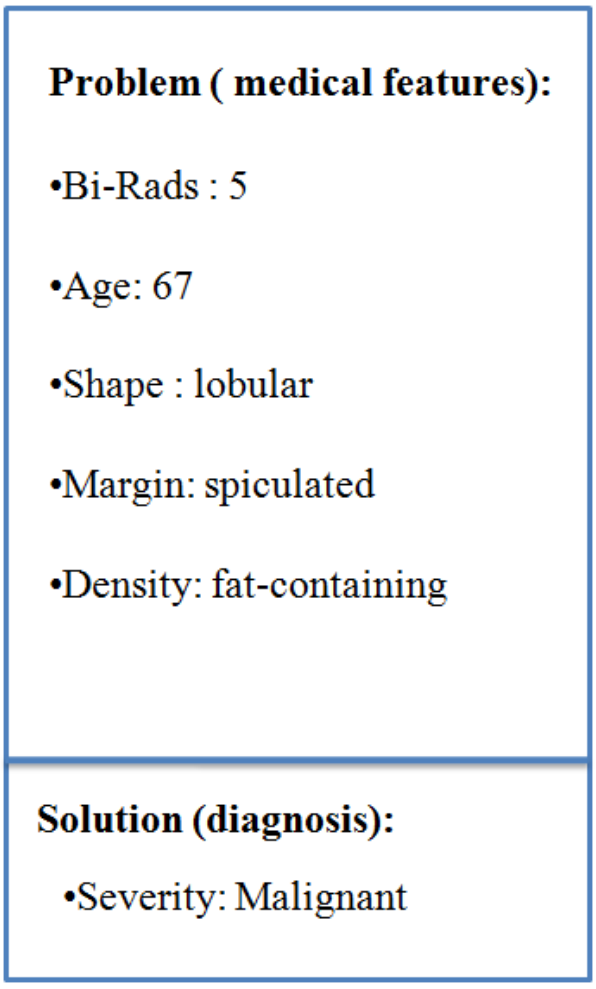

Figure 1.5 – *Example of a case structure adapted to the diagnosis.*

1.4 CBR life cycle

The number of phases that make up the CBR life cycle may vary depending on the different literature resources. Aamodt and Plaza[5] where the first author to have described the CBR life cycle, it defines the process of solving a new problem by following the four steps known as «4 REs »:

1. **REtrieve** is the process of searching for similar case(s) within the case base,
2. **REuse** is an attempt to solve the problem by adapting the retrieved case(s),
3. **REvise** involves the evaluation and repair of the selected case(s), if the proposed solution is inadequate this process can correct it,
4. **REtain** is the process of learning, it enables the CBR to learn and create new solutions; the new resolved case are be added to the case base for future use.

It is also common, in several application areas, to find CBR systems that are able to retrieve appropriate knowledge, but they leave it to the user to determine an interpretation of the final decision produced[24]. In these conditions, the reuse and revise steps are not implemented. In fact, the retrieve step is used solely to support the reasoning task.

The classical 4REs model of the CBR life cycle according to Aamodt & Plaza is presented in Figure1.6.

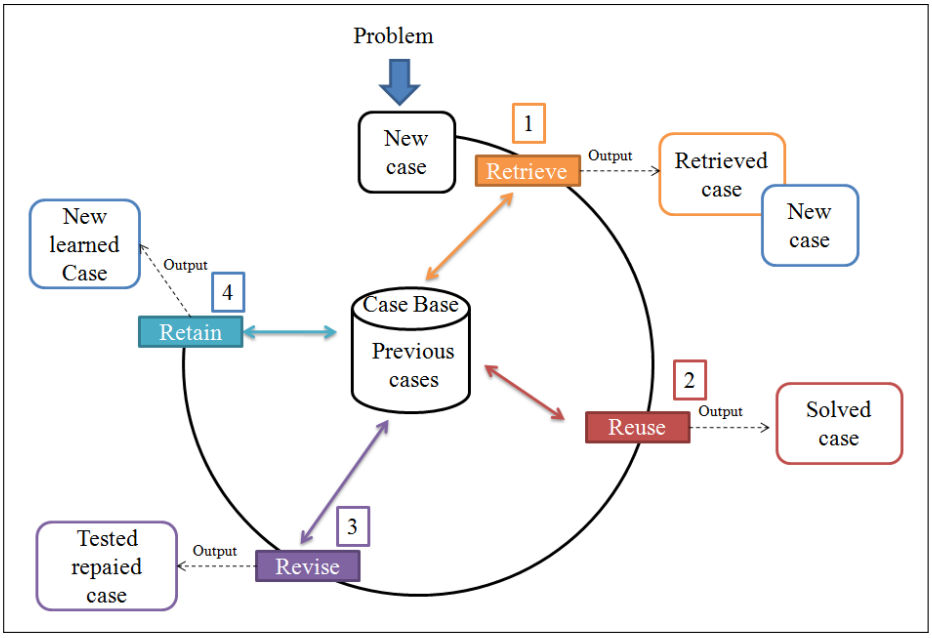

Figure 1.6 – *Case Based Reasoning life cycle [5].*

Later, Mille [6] introduced a gentle modification to the cycle by adding a preliminary Elaboration phase to the beginning of the life cycle. The Elaboration phase is the process in which the target case is built by completing or filtering the description of a problem from a possibly incomplete description. Figure1.7 shows the CBR life cycle with the five phases.

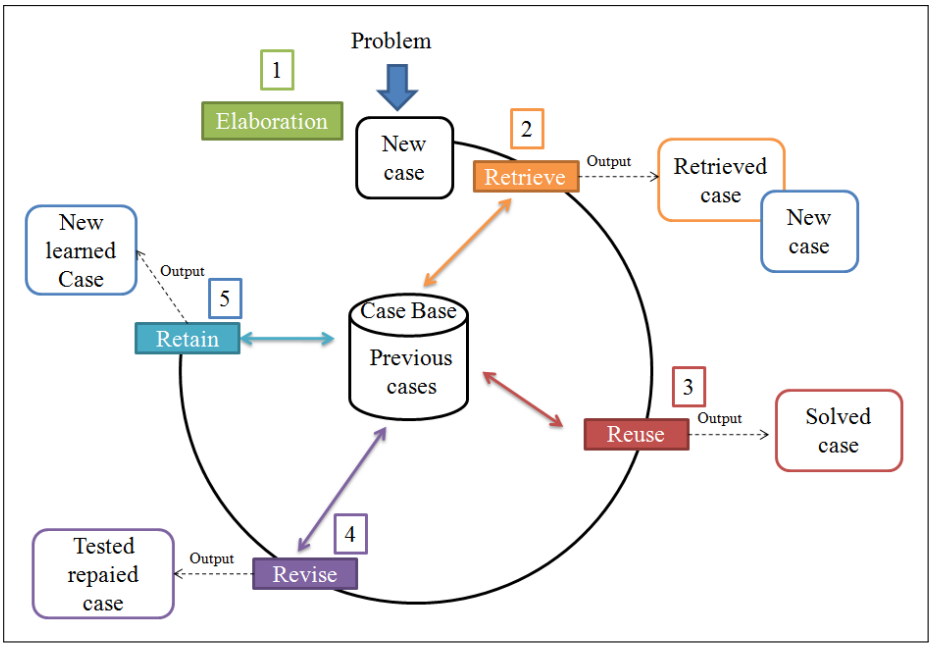

Figure 1.7 – *Case Based Reasoning life cycle [6].*

Each of the 4RE phases of the classical CBR life cycle presented by Aamodt and Plaza [5] will be discussed separately in the following subsections, as well as other phases proposed in literature.

1.4.1 Application phase

Reinartz et al.[41] regrouped the three first steps of the CBR life cycle, namely: retrieve, reuse, and revise phases under the appellation of Application cycle(phase). A revisited version of the 4REs is proposed, as the model is deemed to provide a complete description of the running system, but regarding its maintenance, the model is insufficient. The CBR life cycle is extend and decomposed into an Application and Maintenance cycles (phases) (see Figure1.8).

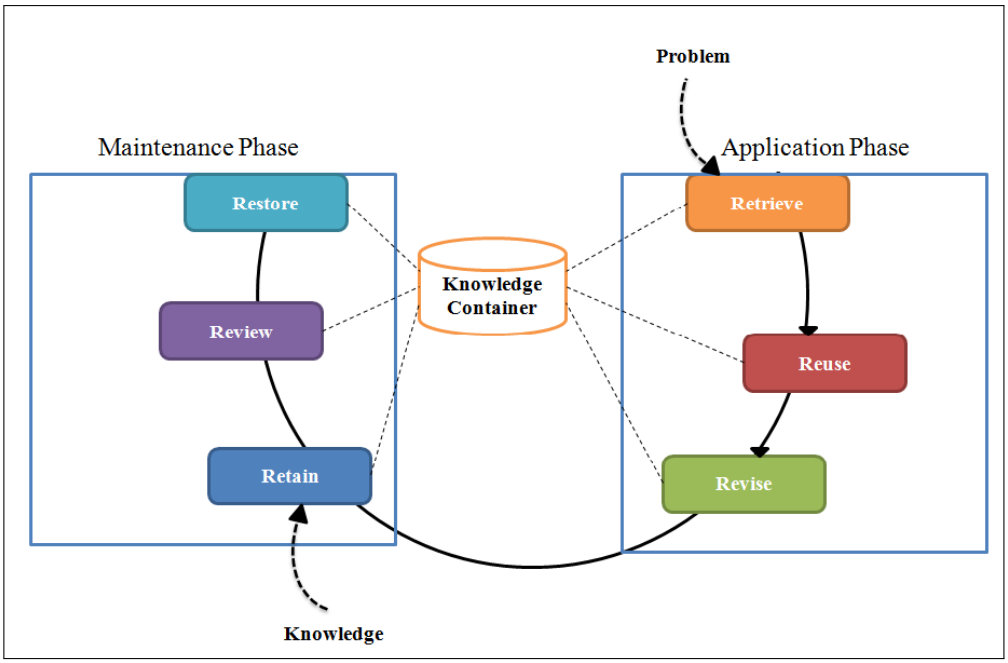

Figure 1.8 – *Six REs cycle (Application and Maintenance phases) (Adapted [7]).*

Unlike the four steps model, the six steps model allows new cases to be added to the CB in two ways:

- As a problem introduced to the system at the retrieve step,
- Other knowledge entering at the retain step.

CBR methodology consists of finding a solution to a given a new problem by retrieving and reusing previous experiences stored in the CB. The problem-solving episode involves the *retrieve* and *reuse* stages of the CBR system, as well as some feedback (evaluation) that can be obtained in the *revise* phase, where a remade solution is generated and annotated with the evaluation outcome[3], as shown in Figure1.9.

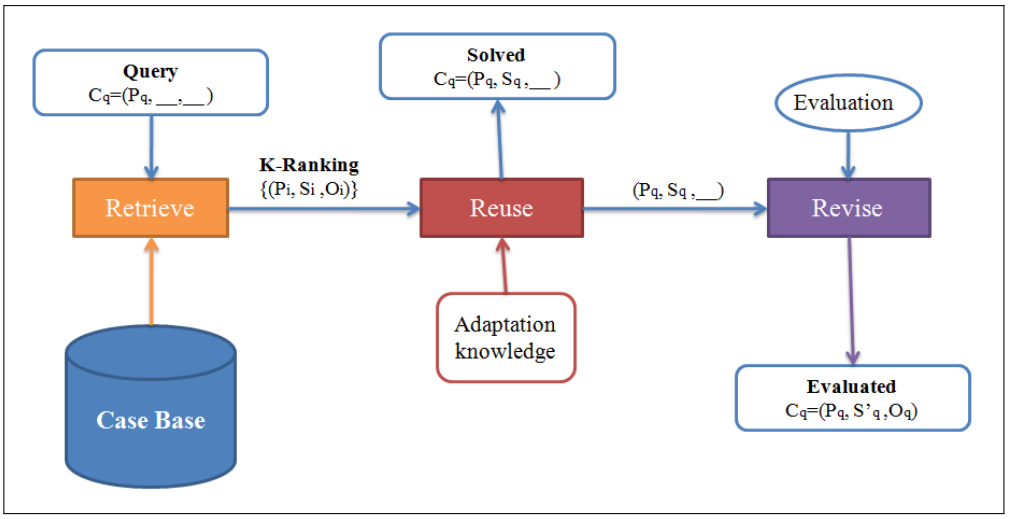

Figure 1.9 – *Reasoning and decision in CBR[3].*

1.4.1.1 *Retrieve*

The intention behind any problem-solving method is to obtain a good solution, ideally the best solution [4]. The question that requires an answer is «*What case in the CB has the most suitable solution that can be reused to solve the new problem?*»

Given the description of a query problem, a retrieve algorithm should search for most similar cases to the current problem using its index. The retrieval algorithm depend on the indexes and the organization of the CB to target potentially useful case(s). The following tasks should be performed:

1. Feature identification: define the problem and deduce the problem description,
2. Search the CB to retrieve case(s),
3. Matched retrieved case(s) according to a similarity measure,
4. Select the most suitable case(s) for the query case.

Once similarity is calculated between the query case and cases in the CB using similarity measures, ranking and selection tasks take place. As the retrieve phase turns a set of cases with various similarity degrees to the query Cq, the most suitable case(s) should be selected for use in the reuse phase (Figure 1.9). As a result , cases are ranked given a preference relation induced by their utility for solving a case[3].

Two approaches are considered in the retrieve phase [1], those:

- Based on the calculation of the similarity between the source case and target case,
- Using in addition to the notion of similarity the notion if diversity.

The objective of the first approaches is to find case(s) within the CB that are similar to the current problem, by measuring their degree of matching, in the sense that they are easily adaptable to this new problem. As for the second type of approaches, their objective is to recall case(s) similar to the target case, and choose among these cases those that are not very similar to each other. Amid the well known methods for case retrieval are: nearest neighbor, induction, knowledge guided induction and template retrieval. Generally these methods are used along or combined to make a hybrid retrieval strategy.

1.4.1.2 Reuse

Using a case is to reuse a previous experience for a new problem. If the new problem is identical the previous one (retrieved case supposedly successful), then the reuse is simple where we just copy the old solution. The reuse Principe for a selected case is presented in Figure1.10.

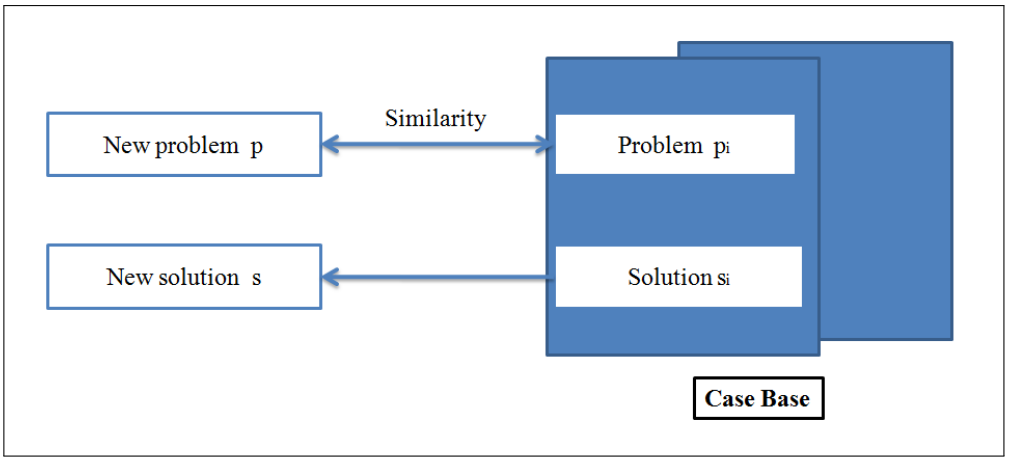

Figure 1.10 – *Abstract of CBR procedure.*

If the new problem differ from the retrieved case, adaptation is required. Fuchs et al.[42] consider adaptation as a plan, whose initial state is the starting solution and the final state is the adapted solution.

It is infrequent to be able to reuse directly the solution as it is stored. This especially happens if the new problem does not differ in essential aspects from the nearest neighbor selected from the CB [4]. It is recommended then to adapt the stored solution before reusing it, in order to best suit the new problem. This phase can be carried out either through human intervention (manually),or automatically using algorithms, methods, formulas, rules, etc.

1.4.1.3 *Revise*

Revise phase begins when a solution is proposed to solve the new problem, and it is complete when the solution is confirmed. The aim of this phase is to evaluate the relevance of the proposed solution at the end of the reuse phase[4]. The revise phase includes solution evaluation, along with the solution repair, if needed.

This evaluation concerns several actions that can be employed[6], it can be done in real world directly through an evaluator, or indirectly by calculating, for example, a measure for certain conditions in the application domain. Generally, a loop between reuse and revise is regularly performed, until the correct solution is obtained. Moreover, at the end of the revise phase, additional information could be gathered (explanation of failure for instance) and saved for future problem-solving improvements[3].

A sub-process of the application cycle is illustrated in Figure1.11, where the three steps along with the task decomposition of each one is presented.

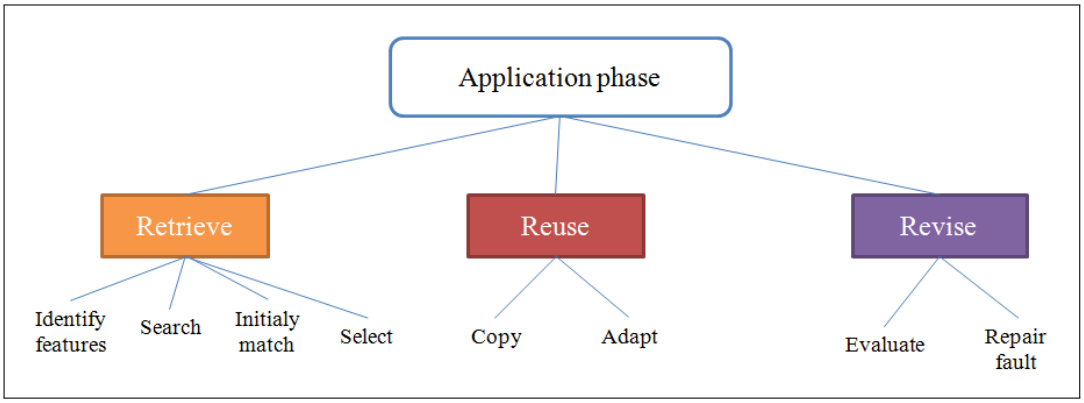

Figure 1.11 – *Application phase[7].*

None of the application phase steps introduce any changes on the knowledge container of the CBR system, it is just the phase where a user solves a problem using previous experiences already stored in the CB. ‘

1.4.2 Maintenance phase

During maintenance, the CB quality is monitored using machine learning methods that allow the application of a maintenance strategy. Maintenance is commonly defined in software engineering and knowledge engineering as an activity that takes place after the development of the system is completed, and the application has already been deployed to exploitation [3] (see Figure1.12).

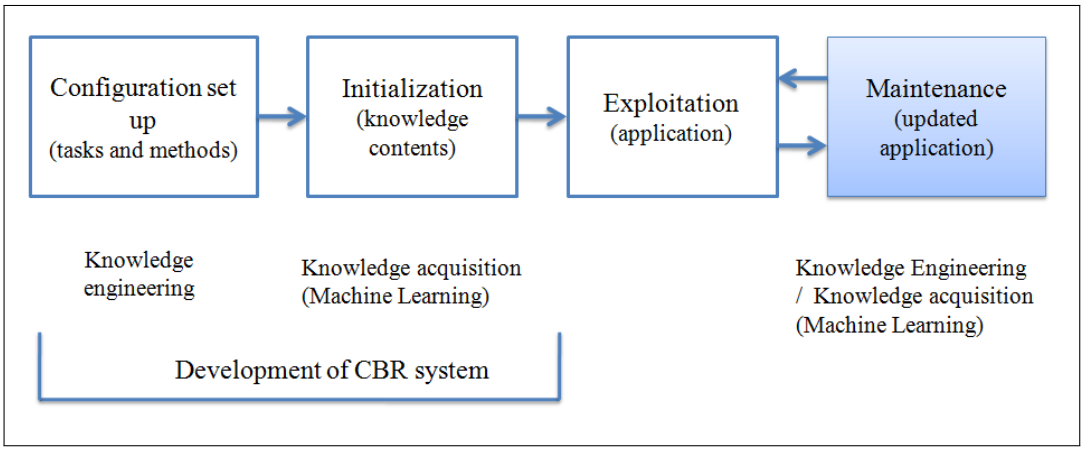

Figure 1.12 – *Processes of CBR system(development, application and maintenance).*

Maintenance consists of introducing changes on the system, for the purpose of correcting faults, improving performance or the system features, and also to adapt the system to eventual changes in the environment in what is known as: Corrective, Perfective and Adaptative maintenance, respectively[4]. In the original CBR life cycle proposed by Aamodt and plaza[27], the maintenance step consists only of the retain phase[7].

1.4.2.1 *Retain (According to Aamodt & Plaza [5])*

Retaining cases in the CB is the point that enables changes to be introduced into the CBR process through the new solved cases (description of the query and the revised solution) stored in the CB. This phase consists of incorporating what is useful to retain into the case base and synthesizes new knowledge that will be reused later.

The storage of a new case thus allows to enrich the case base allowing the increase of the system experience. The decision to either retain a case or not in the memory relies on whether a CBR system has some introspective maintenance policy as analyzing the state of the CB and future problem-solving or for instance the utility problem(deals with the size and retrieval time of a CBR system). At this particular step, the CBR begins the maintenance phase[3].

However, maintenance can be computationally expensive which decreases the system efficiency if it is executed during every application cycle[43]. To solve this, a separated maintenance cycle was introduced by Göker and his co-authors[8], which was then developed by Reinartz et al.[7] to the six steps cycle. The maintenance cycle proposed by Göker et al.[8] runs when a particular condition is satisfied (for instance the CB reaches a predefined limit), or when a knowledge engineer deems that maintenance is necessary[44].

Göker and his co-authors's maintenance cycle[8] is presented in Figure1.13, every time a new problem arises, the application cycle (Retrieve, Reuse and Revise steps) is performed normally.

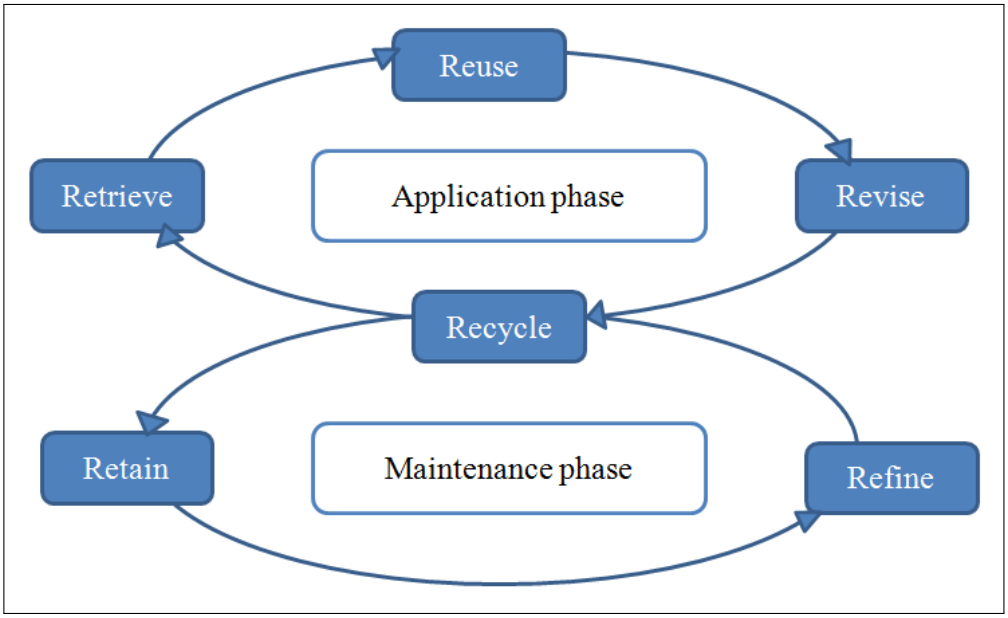

Figure 1.13 – *The CBR cycle proposed by Göker et al[8].*

The next step in the cycle is the **Recycle step**, it encompasses two functions, first of all it is a continuation of the application cycle, as it puts the generated solution to use [43]. Second, if the system generates an incorrect solution, and the user generates a new one, it is stored in a buffer and send to the maintenance cycle. Retain and Refine are the steps that constitute the maintenance cycle:

1.4.2.2 *Retain(According to Göker et al.[8])*

In this step, the CBR administrator checks the new cases are checked for quality. The cases must have correct, relevant and applicable solutions.

1.4.2.3 *Refine(According to Göker et al.[8])*

In this step, the aim is to optimize the CB performance by refining this knowledge container, in order to keep it correct, with a maximal coverage and no redundant cases. The cases that were examined for quality in the retain step are now examined to see if retaining them in the CB may cause redundancy or inconsistency.

The proposed six steps CBR model presented by Reinartz and his co-authors [41] is another variation of the CBR life cycle. An extension of the original four steps model was proposed, by adding two steps into the maintenance phase, namely, *Review* and *Restore*(see Figure1.14). The revised model (see Figure1.8) emphasizes the importance of maintenance in modern CBR.

1.4.2.4 *Review*

The review stage consists of measuring and monitoring the CB (Figure 1.14) as a consequence of the actions performed in the retain stage. This step consists in considering the actual state of the memory (CB) and assesses its quality[7]. Including new cases in the memory can degrade the efficiency of the systems, to this end, appropriate quality measure are used to indicate the quality of the assessed CBR system. Several measure are considered for this task, they are generally grouped under: Syntactical measures and sementical measures[3].

Syntactical measures such as consistency, uniqueness and minimality[7] are not based on the knowledge domain, except for correctness, which is directly related to domain theory. In contrast to the syntactical measure, semantical measures as case density or CB distribution use domain knowledge. Coverage and reachability are also very well-known used quality measure (these points are discussed in more detail in Chapter2).

The use of quality measures serves as monitoring operators that allows a continuous control, as specific indicators lead to the initiation of the *restore* step[7]. These measures are useful especially when a degradation in the quality is noted, the review step suggests changes that can help bring the quality back to the desired level.

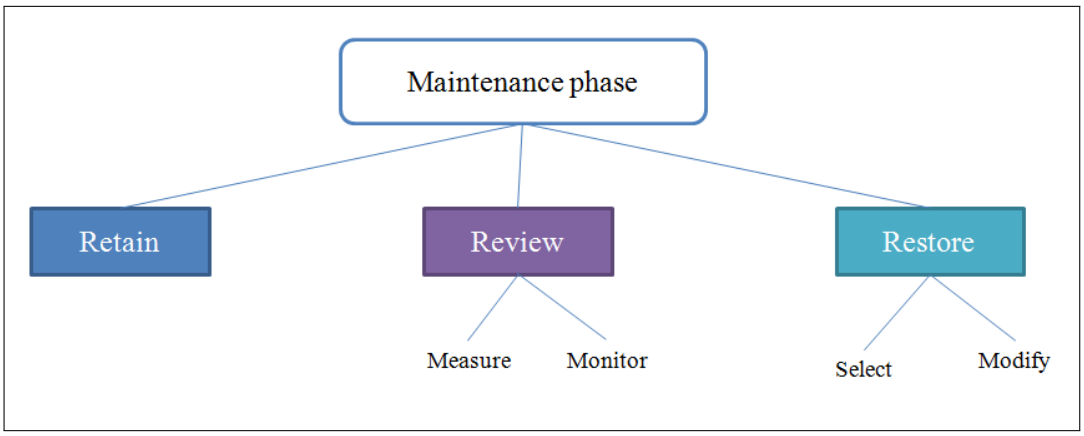

Figure 1.14 – *Maintenance phase[7].*

1.4.2.5 *Restore*

Restore step encompasses two tasks (Figure1.14), Select and Modify. When the review steps indicates that is necessary to get back to a desired quality level, operators are selected and used to modify the content of the CBR system. A maintenance algorithm is used to decide which operator is used to restore the desired quality, several operators can be used in the restore step: addition, deletion, specialization or generalization,etc [43].

1.5 Application domains of CBR

CBR algorithms have been successfully applied to a large range of tasks in several domains including the following[38],[9]:

- legal reasoning,
- health science,
- industrial, juridical and financial analysis,
- maintenance,
- real estate appraisal,
- forecasting,
- etc

Very interesting papers are published in the book «Successful case-based reasoning applications »[24], the book collects a set of excellent papers on CBR application in many fields. Even after fifty years of existence, CBR continues nowadays to be employed for the implementation of applications in several areas, for instances: emergency decision making model for environment emergencies[45], energy optimization [46], real estate evaluation[47], deciding solution of mechanical failure of a car[48], monitoring elder people[49], diagnosis of gastrointestinal cancer[50], detection and classification of nosocomial infections[51], bankruptcy forecasting[52],telemedicine[53], ect

Several types of applications are also found, namely, knowledge management, planning, decision supports, classification and diagnosis. According to authors, typologies of applications of CBR are proposed, they depend on the field addressed and the nature of the task to be carried out.

1.5.1 When to use CBR technology?

CBR is a methodology used to develop knowledge based systems. Althoff[9] defined characteristics of a domain to distinguish whether CBR approach would be applicable :

1. Existent records of previously solved problems,
2. Problems are repeated and the experiences learned may be essential in the future, they are considered an asset that must be preserved,
3. Remembering previous cases becomes intuitive when lacking case history,
4. Examples are given by specialists when discussing the domain ,
5. It is acceptable to use approximate solutions.

The last condition is more of a necessity (requirement), as opposed to the first points which may not all be met. The last point helps provide more or less appropriate (similar) solutions. If all the necessary conditions are met, then the implementation of a CBR system is plausible and the study of the field of application can begin. This study therefore depends on the nature of the application, its characteristics and the objectives to be achieved. The CBR systems differ between them by the formalization of the case, the use of knowledge models, the stages of the cycle and the different algorithms and methods used in each stage.

1.5.2 Typologies of applications

Among the very first categorizations, one has been proposed by Watson & Marir[38], the authors divided the applications according to the type of use: commercial and academic applications. While Althoff and his co-authors[54] decompose the applications according to the type of tasks to be performed.

Althoff[9] proposed four hierarchical levels by introducing the notion of hierarchy of applications in relation to the complexity of problem-solving (see Figure1.15):

1. **Case-based classification** is at the first level of the hierarchy in which the solution of the problem is related to the selection of one or more classes;
2. **Case-based diagnosis** comes just after, it is considered as a generalization of classification. This generalization concerns the general knowledge of the domain that will search for necessary information (symptoms) in a diagnosis process;
3. **Case-based decision support** is on the third level of the hierarchy and distinguishes symptoms of problem solving which are not always direct or visible;
4. **Case-based knowledge management** is a more general and complex application, in which no method or reasoning can be directly applicable.

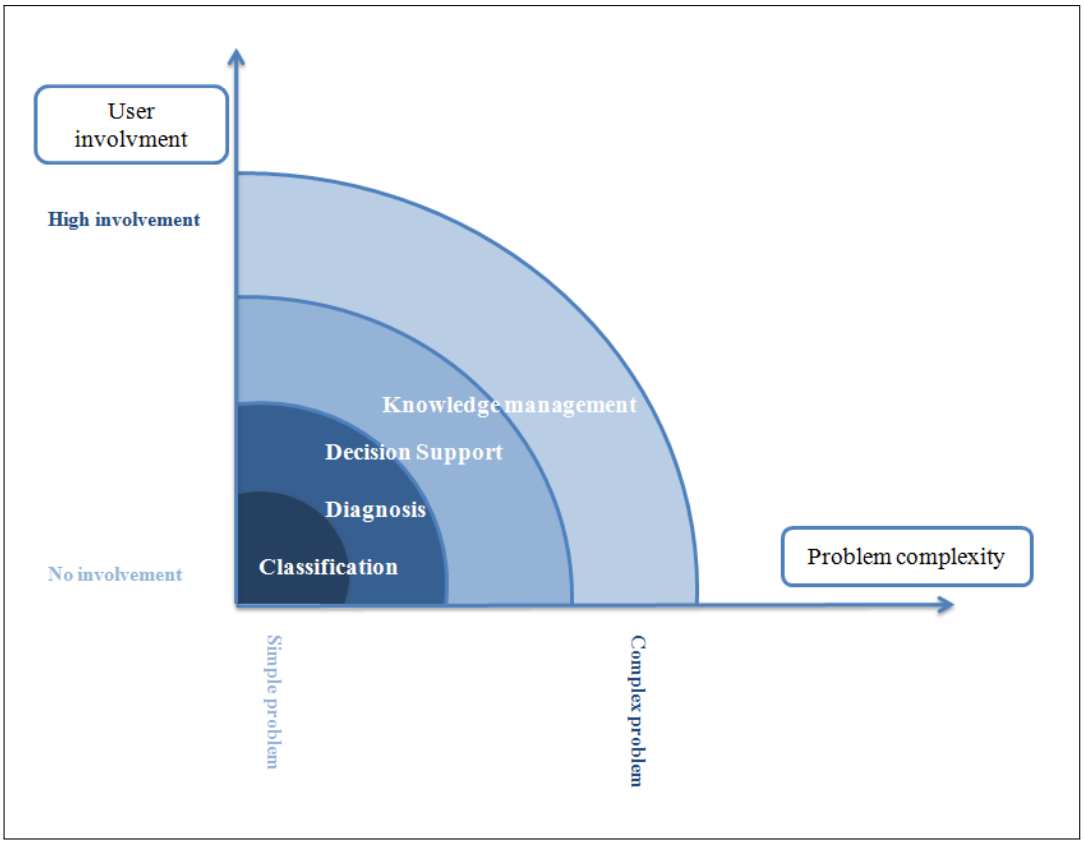

Figure 1.15 – *Hierarchical levels of CBR system's application according to [9].*

1.5.3 CBR in Medecine

What drew our attention to the CBR approach was the remarkable success that this application has had once implemented in the field of medicine, and the number of CBR systems proposed in the field to date, for instance the COVID-19 diagnostic system proposed by Smiti and Nssibi [55]. The reason that motivated our research question is therefore the amount of important work done in the field of medicine, and subsequently we searched a little deeper, focusing on one of the primary steps for the implementation of CBR systems which is the acquisition of the case base.

Therefore, we have decided to dedicate a section of our thesis to summaries in Table1.3 some of the well-known CBR systems in medicine. A more elaborated survey is presented in [56] ,[20] of medical CBR systems proposed in the past few decades.

Table 1.3 – *CBR systems in medecine.*

Name of the system	**Objective**	**Year,Ref**
SHRINK	diagnosis of psychiatric disorders	1987,[57]
PROTOS	diagnosis of auditory disorders	1987,[58]
CASEY	diagnosis of cardiac disorders	1988,[30]
Florence	diagnostic, pronostic et prescription de soins infirmiers	1993,[59]
MERSY	taking care of workers' health in rural areas	1995,[60]
MacRad	interpretation of medical images	1998,[61]
KASIMIR	treatment of breast cancer	2000,[62]
FM-Ultranet	diagnosis of fetal deformities	2003,[63]
ICONS	antibiotic treatment	2007,[64]
RespiDiag	Diagnosis of Chronic Obstructive Pulmonary Disease	2014,[56]
CEDS	Diagnosis of Cholera	2015,[65]
BTCBRsys	Breast cancer diagnosis	2017,[66]
/	Diagnostics of Cardiovascular Diseases	2020,[67]
/	COVID-19 Diagnosis	2020,[55]

1.6 Conclusion

Amid various Artificial Intelligence tracks, Case based-Reasoning (CBR) mimics the human reasoning process. Hence, it is a methodology that has been very promising in various domains for several tasks. In this first chapter of our thesis, we have presented the research axis that inspired our research topic, namely CBR. We discussed the essential points concerning this methodology, starting with the fundamentals necessary to familiarize with the different terms of the approach. Thereafter, the structuring and representation of the case were presented, along with the case indexing and the case base organization. Afterwards, the CBR life cycle was discussed along with its detailed steps that allow the manipulation of this knowledge mechanism. Finally, we concluded this chapter with some CBR applications, discussing how CBR can be used, and the characteristics that allow this approach to be employed in certain domains. The success of CBR even after

forty years of existence, and the fact that it is still today a highly rated artificial intelligence approach is the reason we have been interested in this methodology. Particularly, we are interested in the "heart of CBR" systems, namely, the case base, the acquisition of knowledge, and its maintenance. As can be seen in the conditions of application of CBR in any domain, the first point is the existence of records previously solved problems[9].Thus, this associates the learning aspect of any model to its case base (or the training set). Yet, we face a major obstacle when it comes to CBR systems, particularly for medical applications, namely the difficulty of assembling labeled case bases, traditionally assumed to exist or determined by human experts a point that we will develop in the following chapters.

Maintenance of the CBR system

Contents

2.1 Introduction

Case-Based reasoning systems are implemented to operate over a long period of time, supporting active learning of new cases through to the retain step of the classical CBR life cycle. A long term CBR application eventually lead to an acknowledgment of the importance of maintaining the case based reasoner.

The retention of cases in CB at the end of each CBR life cycle leads to a very rapid growth of the CB, which can negatively affect the quality of CBR, and results in a slow execution of queries in the retrieve phase. This performance degradation is due to memory swamping or the exposure to harmful experiences[68]. Both factors affect the general utility of a CBR system. The swamping problem is related to the cost of searching in a large CB for suitable cases to solve the current problem. Meanwhile, the harmful experiences assert that some cases within the CB may degrade the performance of the system, namely, irrelevant, incorrect and redundant cases.

To avoid performance degradation, CBR systems must be maintained, and various works have been proposed to cope with the mentioned challenges. All with the same goal, namely to insure and enhance an efficient CBR process. The proposed enhancement target different parts of the CBR course, and are divided into : maintenance policies and integration of maintenance with the CBR process. Some studies centered their attention on the reasoner part (e.g., defining two steps Review and Restore to integrate maintenance in the CBR life cycle[7]), others focused on the CB being the essence of the learning for the CBR system. In a work presented by Iglezakis et al.,[69] the author recommends the CB scanning, as it is the knowledge container in CBR systems. The CB is sensitive to changes and its consultations is important to activate the maintenance activities. This maintenance is manifested by a set of possible actions, such as the deletion of irrelevant cases, selection of groups of cases enabling the elimination redundancy and improve the reasoning power of the system, in addition to rewriting cases to repair

inconsistencies[11]. Moreover, CBM can begin with an analysis process[44], this process allows the maintenance operations to start, and it can be done «online»or «offline». These maintenance operations are based on CB quality assessment criteria.

A key feature of CBR is that when it comes to learning, the latter is incremental and continuous. Newly solved cases are added to the CB (retain step), thus the learning is involved in the problem-solving process itself. We begin this chapter in Section 1 dedicated to learning in CBR, particularly learning related to the CB knowledge container.

According to Oxford Learner's dictionaries, maintenance is the act of keeping something in good condition by checking or repairing it regularly. It has already been mentioned before, maintenance is generally defined in software engineering and knowledge engineering as an activity that takes place after the development of the system is completed, and the application has already been deployed to exploitation[3].

Accordingly, it is natural to discuss the development of the system first, before moving on to system maintenance. In this work we draw attention to the fact that most existing CBR papers focusing on the life cycle of the system once it is operational, but rarely on the stage of development or the problems that can be encountered when trying to develop a case based reasoner, namely, the acquisition of data to build a CB for CBR, as it is a crucial step for the development of the system, as this knowledge container represents the heart of CBR. In section 2, we discuss the development phase and maintenance process, along with the main maintenance activities and the framework to categorize maintenance polices. Understanding the nature of maintenance process and how it is related to the overall CBR process is advantageous for identifying good research opportunities and appreciating maintenance practice. In section 3 we detail the concept of case base maintenance, and discuss the different quality criteria for the evaluation of the CB, in addition to three different categorizations of CBM policies found in literature.

The issue of maintenance arises when considering the development of case-based reasoner, a support tool is required to monitor the state of the system and to determine whether, when and how to updated the system's knowledge in order to accomplish the performance goals set. Section4 in a related work section where we present and discuss different CBM algorithms, how they operate, the direction of the approach and summarized the different strategies in a table for a better reading guided by a comparison according to some key characteristics.

2.2 Learning

Learning is a process in which a constructed representation of experience is constructed [70]. Maintenance is a process in which that organized representation may be subject to change, which makes learning and maintenance complementary. CBR is a methodology for both reasoning and learning, the reasoning part was detailed in Chapter1, as for learning, it can be performed at different stages of CBR life cycle. Although CBR has been defined as a lazy learning approach[3]due to the fact that no learning effort is committed when storing cases in the memory, but dedicated at problem-solving time when reusing the case, a more practical case-based learner necessitates eager methods for an efficient problem-solving.

A learning can start with a simple retention of a case after revision, to a more advanced machine learning algorithm or CBM strategy. However, the way of learning changes according to the stage at which it is applied, i.e., there is a difference between the learning at the revision step of the life cycle and while maintaining the case based reasoner. The former is a local step, is deal with only one case with respect to a given query, the policy usually employed is to change the case as little as possible[4]. As for the last one, it is more global, maintenance has influence on the CB in its entirety, as it deals with more global aspects than just one case as in the revision step.

The purpose behind learning is improvement, it is always associated to the system's development, application and maintenance and performed at any of the three stages, it is applied to any knowledge included in the case based reasoner, i.e., the knowledge container. In this chapter we are particularly interested in learning during the maintenance phase, to enhance the quality of the CB.

2.2.1 CB container

One of the essential characteristics of CBR is that when it comes to learning, the latter is incremental and continuous. Newly solved cases are added to the CB (retain step), thus the learning is involved in the problem-solving process itself. Still within the framework of general learning, we recognize two contradictory demands on the CB:

- A CB should be well informed to supply good solutions to any given queries,
- A CB should be small for an efficient retrieval.

The aim of this learning is to improve the CB, for an efficient reasoning. Yet, to acquire a well informed CB, new cases resulting from the revise step are retained if this seems to be appropriate, and this may even be disadvantageous for the CB as it leads to an uncontrolled growth of the storage size[4]. In this view, the retain stage is considered as a place for more involved learning activities. To be able to achieve learning's demand, some useful properties then must be defined (Section Quality criteria for CB evaluation).

2.3 Development and maintenance of CBR system

Most CBR work focuses on the short life cycle of the system once it is operational, but rarely on the stage of development and the problems that can be encountered when trying to develop a CBR system. Development and maintenance of any project are closely related [4], moreover, development is considered maintenance on a zero basis. Development is not discussed in the CBR life cycle, yet, when dealing with CBR, the system development is the primary step one has to perform. Closely related to maintenance, development also involves filling and structuring knowledge containers, aiming to improve the system. The only difference is that development takes place first, followed by maintenance once the application has already been deployed to exploitation[3].

2.3.1 Development

Several reports describe how CBR system have been developed, it is recognized that this is a creative task. This is analogous to the way computer programs were developed in the early days[2]. The steps of the development of a CBR systems are similar to the steps of development of any other system, the process is highlighted in Figure2.1.

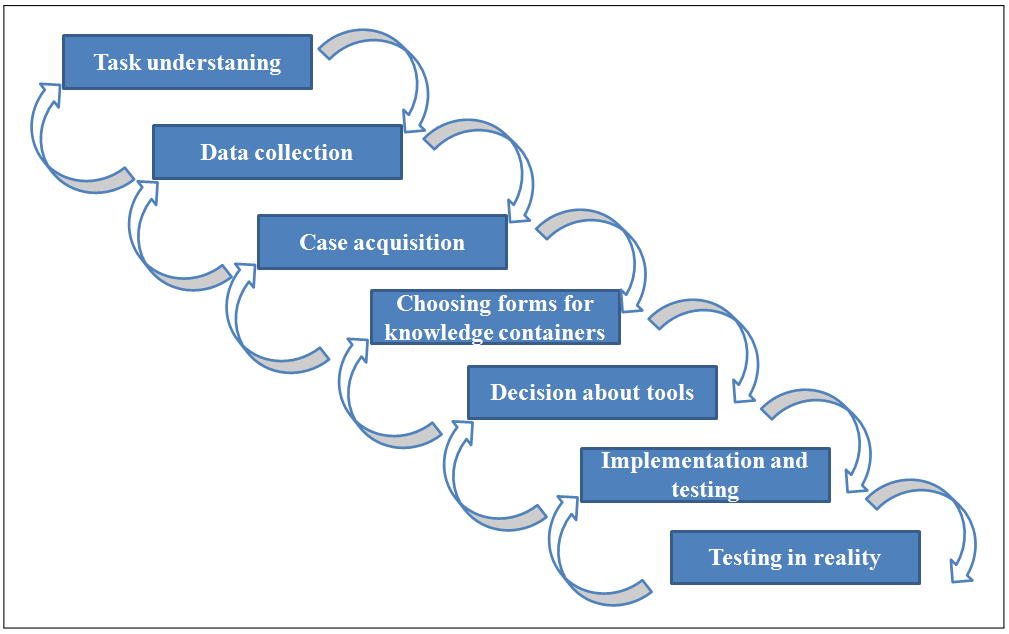

Figure 2.1 – *Steps in system development(adapted from[4]).*

.

2.3.1.1 *Data collection*

It is important to emphasize that there is no entirely automated method for building a CB, of course there is always exceptions, as situations where the cases come from a preexisting CB or a data base. Cases are extracted from the raw data once processed and refined, thereafter they are structured to have the form of a case, that is:

- Queries;
- Solutions.

It is possible that the data available from the given raw data may not be sufficient, for instance when one wants to implement a Computer-Aided Diagnosis(CAD) system for diseases whose conditions are difficult to diagnose using a CBR framework, or simply when one wants an effictive CAD system, a valuable CB is crucial. For this reason it is necessary to consider additional data extracted from other sources [4].

2.3.1.2 *Case acquisition*

The development step we are most interested in are the second and third step, namely, «Finding and getting data»and «Case acquisition». The acquisition of data to build a CB for CBR is a crucial step for the development of the system, as this knowledge container represents the heart of CBR.

From this point on, it becomes imperative to question data acquisition, which sometimes seems problematic, and to look for a way to build and maintain a quality CB for an efficient CBR system.

The problem of data acquisition is especially but not solely met in fields like medicine for applications such as CAD systems, as this data need to be supervised. Supervision (labeling) is performed by medical annotators with special expertise to ensure that this knowledge is used for learning tasks: training models to make correct predictions, and to achieve reasonable efficiency for CAD systems. This supervision can be seen as a burden for expert annotators, which is very difficult and time-consuming. This labeled training data represents the CB for a CBR system, and a competent CB is needed to correctly represent the variance of the data space, otherwise the generalization performance of the system will be very poor [71].

Taking the above cited challenge into consideration, one deducts that maintenance can be integrated at the development stage to cope with the problem that arises.

2.3.2 Maintenance process

Understanding the nature of maintenance process and its connection to the general CBR process is advantageous for identifying good research opportunities and appreciating maintenance practice. Indeed, there is more to case retention than simply which cases should be stored and learned from, one quickly realizes the importance of CBM. The success of CBM is not exclusively related to the maintenance policy, but also on how this maintenance is integrated within the overall CBR process.

The revision step is considered an elementary maintenance operation, as it is local, and deals with one case in relation to a given query, but maintenance is much more general. Maintenance of CBR is concerned with:

1. Correction,
2. Improvement of performance,
3. Adaptation to changed environment and changed knowledge.

The issue of maintenance arises when considering the development of case-based reasoner, a support tool is required to monitor the state of the system and to determine whether, when and how to updated the system's knowledge in order to accomplish the performance goals set.

Leak and Wilson[44] took interest in the CBM process, and gave one of the very first analysis of this process. The objective of this analysis is to help determine when and mainly how a case-based reasoning system performs maintenance, this is achievable by the categorization of the CB maintenance approaches according to specific defined policies.

The framework to categorize maintenance polices is described in terms of :

- How to gather data relevant to maintenance;
- How they decide when to trigger maintenance;
- Type of maintenance operations available;
- How to execute the selected maintenance operations.

In the framework presented by Leak and Wilson[44], we can distinguish three main activities (see Figure2.2) :

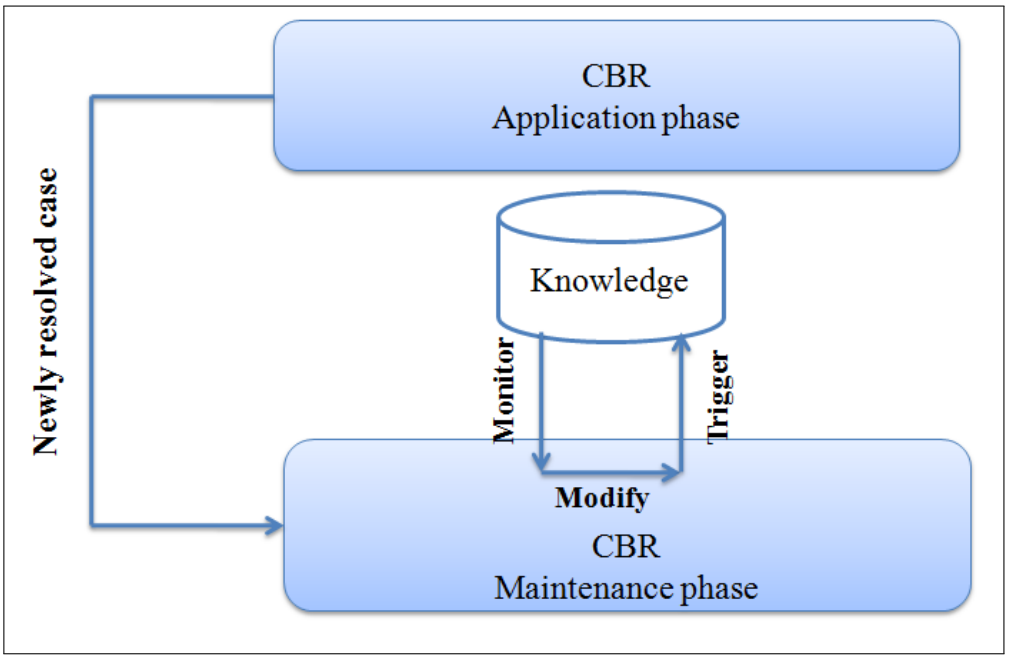

Figure 2.2 – *Maintenance activities (maintenance at the operational level of CBR)[3].*

.

1. **Data collection:** where information about the CB is gathered and analyzed (e.g., how many time a case has been used or how many times a case has been used and presented unsuccessful results),
2. **Triggering:** the information extracted during the data collection stage is used as input to determine whether maintenance is required, and to select the appropriate maintenance actions from a range of possible operations,

3. **Execution:** at this stage, a description of how the selected operation is actually applied to the knowledge container, which leads to a certain modification.

Yet, Leak and Wilson explain that the described framework in their paper "Categorizing Case-Base Maintenance: Dimensions and Directions"[44], is a characterization of a basic combinations of policy attributes. Multiple maintenance policies may be in a single CBR system, each policy appearing at a different stage of the system's maintenance agenda.

Data collection is used to measure the knowledge of a system. A task that can be deployed according to the granularity (type of data), timing, and integration of the measurement with the CBR application [3]. Collecting data means gathering information about individual cases, and/or the whole CB and the general processing behavior of the CBR system. For instance : data collection about an individual case might concern recording the number of times a case is used successfully, or when it has failed.While data collection about the whole CB could concern, for instance, monitoring the size of the CB.

A. ***Type of data*** (also called granularity): three approaches exist to collect and analyze data, this helps deciding when CB maintenance is needed .

   The most simple one is not collecting data at all, these methods are referred to *non-introspective*. These methods collect no data and tend to make maintenance decision regardless of the present or past state of the CB. This is the most used approach in CBR sytems.

   Meanwhile, *Synchronic* approaches have a more sophisticated reasoning. Where they consider a snapshot of the current state of the CB, in a part or as a whole. By examining this information, we can for instance decide if a case is valuable to add to the CB because it increases its competence or if the case in question can effect and degrade the competence[72]. Yet, the most informative approach are referred to as *diachronic*. The latter enables the monitoring

of the CB regarding to changes in the environment. *Synchronic* and *diachronic* are referred to as introspective, since they examine the internal state if the CB.

B. ***Timing :*** the maintenance policy triggering may be periodic, conditional or ad hoc .

Periodic methods have a given frequency to collect data, as for example, collecting data at the end of each problem solving cycle. *Conditional method* follow certain conditions, for instance, they prevent the number of cases in the CB to go over a certain threshold. *Ad hoc timing* is used when an expert or even the user activates the data gathering, for instance, to initiate tests on the CB in order to determine whether maintenance is needed.

C. ***Integration :*** data collection may operate *online* during the active reasoning, or *offline* when there is a pause, for instance while waiting for the user's input .

2.4 Case-Base Maintenance

Case-Base Maintenance is defined by Leak and Wilson[44], as process of refining the CB to enhance the performance of a Case-Based Reasoning system. In [73] CBM is defined as the process of maximizing the quality of solutions of the CB or minimizing its cardinality. In modern approaches, CBM is recognized as another separate step of the case-based reasoning methodology (see Figure2.3). This step aims at supervising the knowledge stored in the CB for a more efficient and accurate reasoning process[10].

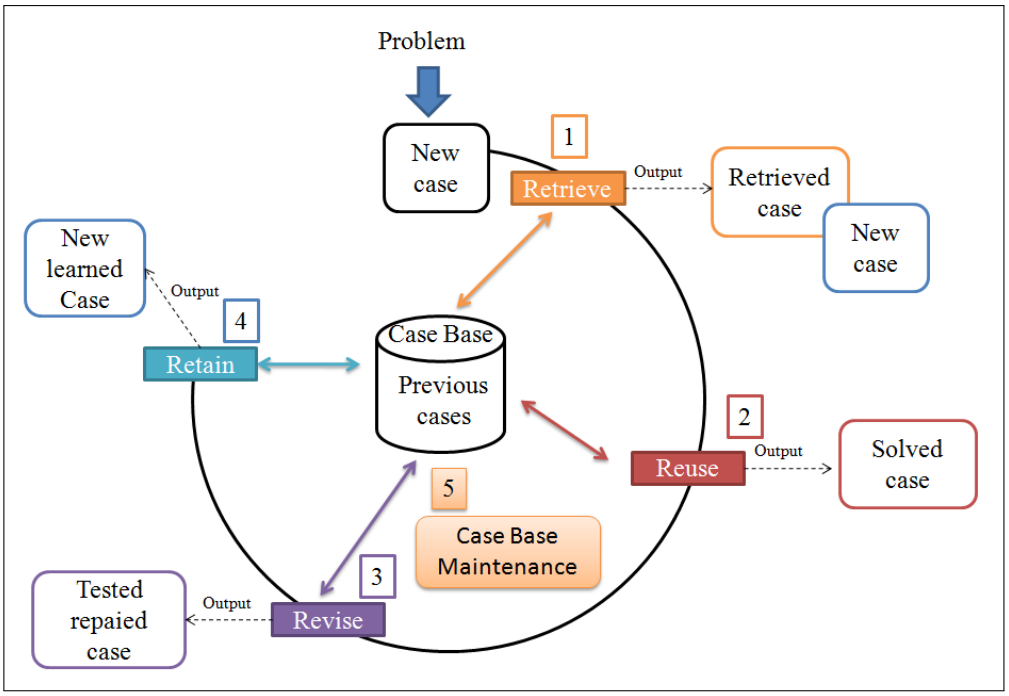

Figure 2.3 – *Case Based Reasoning life cycle: 4 steps and CBM step (Adapted from [10]).*

It is crucial to understand the issues highlighted by a maintenance problem and use this to develop good maintenance strategies in order to assist and improve efficiency and quality of the solutions that the system offers, knowing that its case base is constantly growing and tasks or environment may change over time.

Policies are implemented aiming to ease the reasoning for a specific set of performance objectives, through the revision of the organization or the content of the CB. Revision of the domain information of the CB by adding or deleting cases, or even revising the case representation, changing from a list presentation to a feature-vector representation. Performance objectives are referred as an assessment of the behavior within the an initial CB of a particular CBR system and a sequence of problems solved. These objectives are either *quantitative* (e.g., reaching a precise problem-solving time) or *qualitative* (e.g., enhancing the competence of the system).

2.4.1 Quality criteria for CB evaluation

A CB is qualified effective if it can answer as much queries as possible, efficiently and correctly[40].This evaluation of the CB quality is made according to numerous criteria proposed in the literature: *inconsistency*, *redundancy*, *abstraction* and *relevancy*. However, for the evaluation of the CB two main important criteria are employed ***Competence*** and ***Performance***. *Coverage* and *Reachability* notions are the basis of these criteria:

- ***Competence:*** we measure the competence of a CB by the range of problems it can satisfactorily solve (resolution ability). Thus, the competence of a CB represents the coverage of the case it contains.
- ***Performance:*** directly related to adaptation and results cost, performance of a CB is measure by the response time required for a given query.

Numerous approaches focus on conserving the competence of the CB, a measure indeed based on the two notions of *Coverage* and *Reachability* [72].

- *Coverage:* considered as an important competence property, the coverage of a case refers to the set of target problems that the case can be used to solve.
- *Reachability:* an important property for competence as well as coverage, reachability of a given case, deal with the set of cases that can be used to provide a solution for a particular case.

Given a CB $\{c_1 ,\ldots,c_n\}$ and a set of target cases «T »$\{t_1,\ldots,t_r\}$. A case c consists of two parts: a *problem* and a *solution*.

$$Casec = \{P_s, S_s\}, TargetT = \{P_t, ?\}$$

$$\text{Coverage (c}\in CB) = \{t \in CB : Solves(c,t)\}$$

$$\text{Reachability (c} \in CB) = \{c \in CB : Solves(t,c)\}$$

A satisfactory competence of a CB means that its *coverage* is high and *reachability* low. Illustrated in Figure2.4, we notice that we can remove C_3 without losing competence, contrarily to removing C_2.

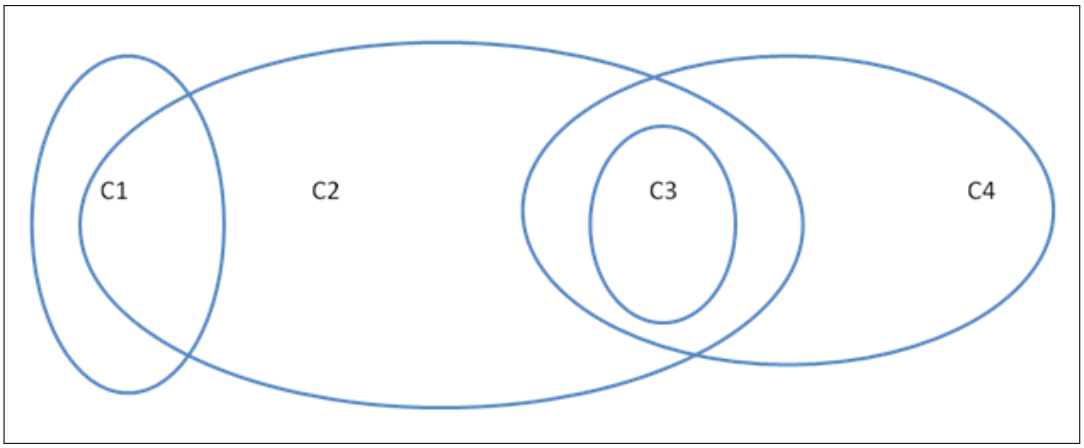

Figure 2.4 – *An example of coverage (based on Smyth and McKenna [11]).*

2.4.2 CBM policies

In literature several strategies have been proposed for CBM, and we can identify three distinct proposed categorization for these CBM policies. In the paper presented by Leak and Wilson[44], a framework for describing CBM policies is discussed. The framework uses a categorization based on **When** and **How** a CBR system is triggered to perform maintenance. The main objective of the different proposed categorization is threefold[44]:

1. Determining classes for similar maintenance approaches, this categorization highlights current exercise in the field and present a better understanding of state- of-the-art CBM approaches,
2. Mapping out what already has been done helps to identify points that were not addressed in previous works, such gaps help to uncover new research opportunities,
3. The categorization scheme for maintenance is a first attempt to catalog approaches according to a distinct performance objective.

Commonly, in recent works, maintenance strategies can be divided into two categories [74]: Competence enhancement and competence preservation, algorithms belonging to the first category refer to methods where noises and misleading information are identified and removed from the CB. While the second category refers to methods targeting redundant cases and removing them from the CB without influencing the prediction accuracy. In the next subsections we discuss three different categorization of CBM strategies.

2.4.2.1 *Optimization and partitioning of CB*

Smiti and Elouadi [75] present a categorization where the entity taken into consideration and studied is the CB. The different presented policies fall under the two categories : Optimization of the CB where algorithms are used to either add or delete cases from the CB [12][76][77]. Partitioning of the CB where a CB structure is build after it is decomposed into small groups of closely related cases[78][79][80]. All strategies in these categories are aimed at developing, optimizing, and maintaining the CB (see Figure2.5).

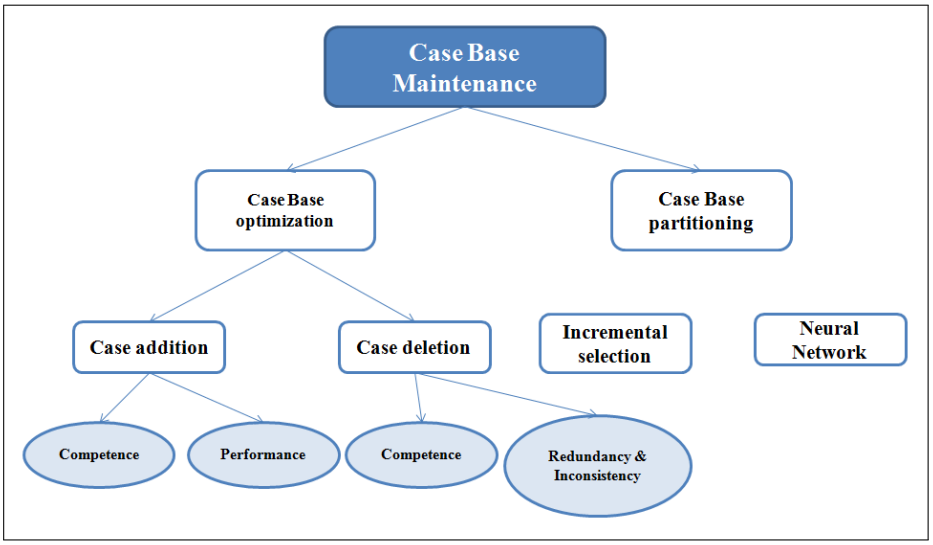

Figure 2.5 – *Diagram of the different strategies and criteria used in CBM[12].*

1.Optimization of the CB: The aim is to reduce the case research time, this may be done following an optimization policy , and employing a case addition(retention) or deletion strategy.

A. **Case addition strategy:** case are added to the CB in order to maximize the competence and the performance, in order to insure a good quality of the CB. Several strategies exist in literature, where researches tend to focus whether on maximizing competence[81],[82], while others maximize the criteria of performance[77].

B. **Case deletion strategy:** In this branch , cases are valued according to certain criteria, for the purpose of suppressing and bringing the CB to a specific number of cases. Different evaluation criteria are proposed, namely, competence, redundancy, and inconsistency (section 2.1). Some of the strategies featured in literature are: random deletion[70], deletion based on redundancy[83], and deletion based on the size of the CB and density[11].

2.Partitioning of the CB: Clustering and feature selection techniques have been successfully applied in the partitioning of the CB[84]. Addition and deletion of cases is possible in each small CB, without using the whole base[78].

Several policies develop a collection of distributed CBs[75],where each element of the distributed CB structure represents on cluster, resulting from the clustering process. These methods are easy to run, as they decompose a large CB into small groups of closely related cases, yet, the completely change the structure of the CB. We may cite methods proposed within the context of partitioning of the CB. Salamo & Lopez-Sanchez[68] have proposed an adaptative CBR mode, the CB is developed during the reasoning cycle by adding and removing cases. In [75] the CB is clustered into small groups, each one is maintained individually, targeting outliers and internal cases. The objective is to reduce the size of each partition while preserving maximum competence.

In [85]an instance reduction method is proposed, Hyper-rectangle clustering algorithm is employed. Subsets of instances near or within the boundaries of classes are selected. The size of the training set is reduced which improves generalization accuracy.

2.4.2.2 *Volume, Variety, Velocity and Value*

In an overview paper presented by Juarez et al.,[13] a selection of CBM approaches published between [2015-2018] are grouped according to their objectives, belonging to four main headings: Volume, Variety, Velocity and Value.

- ***Volume:*** methods for improving the competence model and handling massive data,
- ***Variety:*** methods restructuring the CB, and integrating different data sources to redefine the structure of cases (e.g. removing some feature of the case, or adding new information to the solution part of a case),
- ***Velocity:*** methods managing time in CBM, and providing fast responses while having massive volumes of data to process,
- ***Value:*** methods facing computational complexity, and coping with the challenge of searching for the optimal solution(value solution) without involving high computational cost.

2.4.2.3 *Direct, Hybrid and Case property models*

Another recent categorization is presented by Nakhjiri et al.,[74], where the entity under consideration is the case itself, its behavior either alone in a specific scenario or the relationship it may have with other cases included in the CB. Three categories are proposed:

- ***Direct models:*** mainly all first attempts for CBM strategies fall under this category, where immediate actions are taken upon the classification of the case. No information are extracted from the case nor relationship between the case and other cases in the CB,

- ***Hybrid models:*** are more recent, different artificial intelligence techniques are employed to layout relations amid cases in the CB,
- ***Case property:*** models try to capture the behavior a case in different scenarios by integrating sets and variables presenting additional information of a case, for a better illustration of the characteristics of cases in the CB.

2.5 Related works

Regardless of their categorization, all of the maintenance strategies aim at achieving the same objective: constructing and restructuring of the CB with better quality using different criteria, compared to its initial state. The success of maintenance is correlated not only to the policy itself, but the way it is integrated in the CBR process[23]. In this section, we discuss a selection of different strategies presented in literature between [1968-2020] (See Figure2.6).

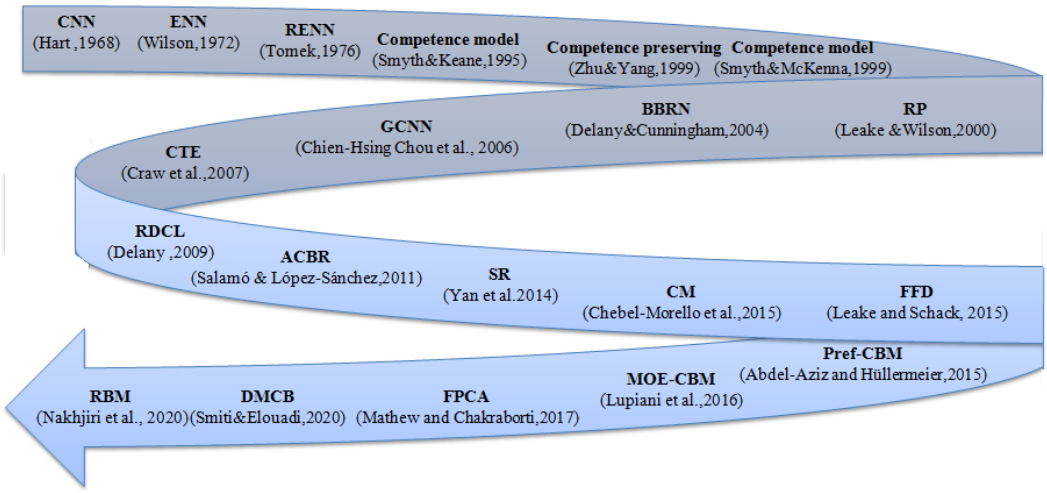

Figure 2.6 – *50 Years CBM: Arc diagram of selected CBM methods [1968-2020](Adapted from[13].)*

A significant number of researchers interpret CBM as a problem of case reduction and proposing algorithms that target noisy and redundant cases[13]. The proposed case reduction methods attempt to increase classification accuracy and improve retrieval efficiency. Researchers then began to take an interest in the machine learning literature and used it as a source to achieve their goals.

CNN& ENN: Different approaches have been proposed to address the case memory reduction , these strategies are based on nearest neighbors(NN) editing rules. One of the earliest attempts in CBM is Hart's Condensed Nearest Neighbors (CNN)[86], and the Edited Nearest Neighbors (ENN) by Wilson[87]. CNN is considered a competence preservation algorithm while ENN is more of a competence enhancement algorithm[74] and they both inspired other algorithms dedicated to CBM.

RENN: CNN is an incremental algorithm, which starts with an empty CB, to which are added the cases misclassified by the other cases currently present in the CB. However, ENN is a decremental algorithm where cases misclassified by their K-nearest neighbors are removed from the CB, as they are considered a noise. Experimental results demonstrate the effectiveness of ENN to increase the average accuracy of CBR techniques. Thereafter, based on the ENN algorithm, a repeated ENN(RENN) method was proposed by Tomek[88].The ENN algorithm is repeated and cases are removed from the CB until no more noise is detected.

Competence model: Thereafter, researchers focused on the competence of the case-based reasoner to solve problems, as case deletion can reduce the later. Smyth and Keane[72] proposed a competence model used for the evaluation of individual contribution of cases. Cases are then categorized according to their competence characteristics which helps guide the selection of case deletion. This competence guided case deletion is considered as a safe way to remove cases from a growing CB, as it saves the systems from the adverse effects of the utility problem[68] while coping with reduction in competence.

Competence preserving: Later on, a competence preserving method was presented by Zhu and Yang[89], the authors use two theories, which aim to find the similarity metrics and adaptation cost to determine the best coverage value to build a CB with near optimal property.

As an alternative to case deletion, Smyth and McKenna[90] used the competence model for a guided case addition algorithm, based on the notions of *Coverage* and *Reachability* to construct a compact competent CB. Using a relative coverage (RC) measure to estimate the unique competence contribution of an individual case. The proposed algorithm adds cases to the CB using the RC metric in combination with CNN to prioritize cases expected to make the largest competency contribution, given their RC value.

Relative Performance (RP): In another paper based on the competence model of Smyth and McKenna[90], Leake and Wilson[77] argued for a more direct integration of performance consideration into case addition. Cases are added based on the performance advantage (PB) they provide through their retention to assess this contribution. To guide the maintenance, a Relative Performance (RP) metric is used in conjunction with CNN algorithm, whose input are ordered by the RP value, assessing the contribution of a case to the adaptation performance of the system.

Blame-Based Noise Reduction (BBRN): Following the same concept as Smyth and McKenna[90], Leake and Wilson[77], the Blame-Based Noise Reduction (BBRN) algorithm was proposed by Delany and Cunningham[91], where for each case a property is measured called *Liability*. The proposed property for a case C is defined as the set of all the cases misclassified because of C. *Liability* is an information integrated into the CB, in the same way as *Coverage* and *Reachability*, to determine the performance of each case under certain tasks or conditions.

Generalized Condensed Nearest Neighbors (GCNN): CNN algorithm inspired so many other works, GCNN is another example[92]. Iteratively samples are selected and others ignored, the ones ignored are the samples that can be absorbed or represented by those selected. The use of a stronger absorption criterion helped to strengthen GCNN, by the addition of a threshold for the difference in distance of the closest class members and the nearest cases to another class.

Complexity Threshold Editing (CTE): Still, within the deletion aspect of maintenance, another redundancy reduction algorithm is proposed by Craw[93]. Since, in classification problems redundant cases are far from decision boundaries in clusters of the same class. Case complexity is used to identify redundant cases with low complexity and boundary cases with high complexity. The proposed Complexity Threshold Editing(CTE) algorithm remove cases whose complexity lies below the complexity threshold.

RDCL: As an extension of the BBNR algorithm, a new paper where another property is presented was proposed called *Dissimilarity*.These four case properties were used : *Reachability*,*Dissimilarity*, *Coverage*, *Liability*(RDCL)[94]. RDCL is an editing method, every case is categorizes based on its properties and its classification by the other cases in the CB. The properties of each case determine which cases should be removed from the CB in order to improve its accuracy.

Adaptive Case-Based Reasoning (ACBR): Salamó & López-Sánchez[68] proposed an adaptive case-based reasoning, taking into account the generational experience (i.e. the history of problem solving episodes) of each case. It is represented by a quality measure calculated over time. In this way, the retention depends on the ability of a case to help the correct classification of other cases, which increases its quality value if no negative feedback is generated.

Selective Retention (SR): One of the factors that reduces the usefulness of a CB is the swamping problem, where the retrieval time exceeds the benefit of accuracy. To overcome this problem, the CBR life cycle was extended in the

work of Yan et al.,[95] by adding a new Refresh phase. From a cognitive science point of view, dynamic maintenance can be enhanced by the selective memory of CBR systems. The memory strategy selectively retains (SR) cases according to two conditions: if the new case is misclassified or if the new case is correctly classified but its similarity to other cases does not exceed a certain fixed similarity threshold, then, the new case is added to the CB.

Competence Measure (CM): Structuring and increasing the CB is also a CB maintenance approach that was proposed, using a competence measure (CM) as in the work of Chebel-Morello et al.,[84]to preserve its quality by taking into account the accessibility and coverage value of both the new problem and its solution.

Flexible Feature Deletion (FFD): A twist in the CBR perspective in suggested by Leake and Schack[96], a flexible feature deletion algorithm is proposed, it enables selective deletion of cases content instead of deleting the entire case, as it causes less competence cause compared o removing the whole case.

Preference BM (Pref-CBM): Abdel-Aziz and Hüllermeier[97] proposed a novel version of CBR named Pref-CBR. In this sophisticated version the classical relation between a problem and its solutions is redefined, instead of having the pair $(Prob, Sol)$ relating a solution Sol to a problem $Prob$, a new notion of *Preference* is introduced, decomposing the case into small blocks of knowledge.

A preference $Sol_i >_{prob} Sol_j$ means that a given solution Sol_i is preferred to Sol_j for solving Prob.In practice this is more of a statistical perspective of CBR[13]. When it comes to maintenance, an algorithm is proposed Pref-CBM to examine whether or not a query case $C = (Prob, Sol, P)$ should be retained in the CB (P is the set of all preference of Prob). A distance function is used to estimate the solution quality, the aim is to avoid retaining a solution along with its preferences are not redundant (already exist in the CB).

Multi-Objective Evolutionary (MOE-CBM): In this next proposed strategy, CBM task is addressed as a multi-objective optimization problem[98]. Lupiani and his co-authors propose to use a multi-objective evolutionary algorithm (MEO-CBM) establishing three objectives simultaneously: (1)minimizing redundant cases, (2)minimizing the distance between the non-redundant cases, (3)maximizing CBR accuracy.

FootPrint Compositional Adaptation (FPCA): The adaptation of a solution (reuse step of CBR life cycle) was considered by Mathew and Chakraborti[99]. Adaptation means that a retrieved solution must be reused (executed) in order to be a valid solution for the presented query, a Compositional Adaptation(CA) is proposed where a graph and nodes are used to represent cases, and the dependency between cases is now considered as part of the solution. For maintenance, a refined version of Relative Coverage(RC)[90] is proposed $FootPrint_{CA}$ to measure the retention score of a case (RS_c) where a high retention score of c means that the case c can solve many cases (compositional adaptation which is the dependency between cases is taken into account for the calculation of RS_C).

Dynamic Maintenance Case Base (DMCB): Smiti and Elouadi proposed a number of maintenance algorithms aiming at the competence improvement of CBR[75],[100], the authors latest proposal is a dynamic maintenance of CB [101]. Using machine learning techniques the CB is clustered into small case bases, and then using a competence model on different types of cases (noisy, similar and detached) some cases are retained while others are removed to reduce the size while preserving maximum competence.

Reputation Based Maintenance (RBM): One of the recent CBM strategies is a Reputation Based Maintenance[74], the authors propose a model to improve the performance of CBR systems. A Reputation value is computed for each case to measure its strength for the classification tasks. The focus is on the performance of the case once retrieved for KNN classification, different variation of the RBM are proposed. For instance RBM_{Cr}, Cr stands for CB reduction where the aim is

to reduce the size of the CB while maintaining its accuracy. Cases judged to be harmful are removed, those case have a Reputation value equal to zero(a case with a Reputation value equal to zero means that it doesn't participate in any classification task).

Finally, a map of 10 CBM methods (recent approaches in our presented literature review) is summarized in the form of a map of some key characteristics (Table2.5), namely: **Data collection**[44] to measure the system knowledge using **Granularity**, **Iiming** and **Integration**(section2.3.2), **Approach** defines whether the maintenance approach uses case/feature editing or partitioning, and last **Direction of CBM** which can be incremental or decremental)[13].

Table 2.1 – *Comparative summary of recent CBM algorithms.*

	Data collection for maintenance			**Approach**	**Direction of CBM**
CBM algorithms	**Granularity**	**Timing**	**Integration**		
RDCL[94]	Synchronic	Periodic	Online	Case editing	Decremental
ACBR[68]	Synchronic	Periodic	Online	Case editing	Incremental
SR[95]	Synchronic	Periodic	Online	Case editing	Incremental
FFD[96]	Non-introspective	Ad-hoc	Offline	Feature editing	Decremental
CM[84]	Synchronic	Periodic	Online	Case editing	Decremental
Pref-CBM[97]	Synchronic	Periodic	Online	Feature editing	Incremental
MOE-CBM[98]	Diachronic	Adhoc	Offline	Case editing	Incremental
FPCA[99]	Synchronic	Periodic	Online	Case editing	Incremental
DMCB[101]	Diachronic	Ad-hoc	Off line	Partitionning	Decremental
RBM[74]	Synchronic	Periodic	Online	Case editing	Decremental

After reviewing some of the maintenance strategies in the literature, we propose a maintenance algorithm at the development stage of CBR systems. In our study we are interested in an incremental direction (retention policy), for two reasons:

- To develop a CBR system $\longrightarrow$ Acquisition of data to build a CB is necessary ,
- Acquisition of data to build a quality CB $\longrightarrow$ A retention policy is needed to retain only valuable cases in the cases base .

We propose an approach to build and maintain a quality CB with minimized annotation cost, using machine learning techniques to overcome the challenge of scarcity of labeled data crucial for the CB (the used techniques are discussed in the next chapter).

2.6 Conclusion

In this second chapter of our thesis, we further explored the field of case base maintenance. Understanding the nature of maintenance process and how it is related to the overall CBR process is advantageous for identifying good research opportunities and appreciating maintenance practice. This allowed us to understand the different courses on which maintenance can be performed, which draw attention to the fact that most existing CBR papers focusing on the life cycle of the system once it is operational, but rarely on the stage of development or the problems that can be encountered when trying to develop a case based reasoner, namely, the acquisition of data to build a CB for CBR, as it is a crucial step for the development of the system, as this knowledge container represents the heart of CBR.

The problem of data acquisition is especially but not solely met in field like medicine for the application of computer-aided diagnosis(CAD) systems, as this data need to be supervised. Supervision (labeling) is performed by medical annotators with special expertise to ensure that this knowledge is used for learning tasks: training models to make correct predictions, and to achieve reasonable efficiency for CAD systems. This supervision can be seen as a burden for expert annotators, which is very difficult and time-consuming. This labeled training data represents the CB for a CBR system, and a competent CB is needed to correctly represent the variance of the data space, otherwise the generalization performance of the system will be very poor.

To cope with this challenge, we decided to implement a support tool, which allows us to build the CB when the acquisition of labeled data turns out to be expensive, challenging and time consuming. Yet, the lack of data can in no way degrade the quality of the CB, because while building it, we maintain it with the help of several machine learning techniques we will present in Chapter 3.

Machine learning techniques

Contents

3.1 Introduction

Machine learning is a field of Artificial Intelligence (AI) that involves the development of algorithms and the implementations of techniques to be intelligent without needing human intervention. The ability to learn from previous analytical observations, and experiences results in a system that can continuously improve, consequently efficiency increases.

In the last decade, machine learning has become very common and popular, and it is used in a variety of applications, namely, automatic recognition of hand writing, computer aided diagnosis, computer vision and speech recognition[102]. When it comes to machine learning, data generally is divided into subsets: a training, and a testing set. The training set is used for the learning of a given model, while the testing set are examples used to evaluate the performance of the learner.

There are different machine learning technique that we differentiate by the type of training data use. A general approach is *supervised learning*, where the training set consists of only labeled data. In contrast, we have *unsupervised learning*, for this approach the training set is made up of exclusively unlabeled data. Halfway between supervised an unsupervised learning, there is *semi-supervised learning(SSL)*, where both labeled and unlabeled data are used to train a classifier(s) such that it is better than a classifier trained on a fully supervised data.

Contrary to unlabeled data, labeled data is often scarce, costly and time consuming to obtain, thus in SSL the small amount of labeled data available is used along with the large amount of labeled data, to reduce labeling cost. In order to select only valuable instances among unlabeled data, Active leaning (AL) has been proposed, it is considered as a special case of SSL and it is viewed as a labeling protocol. The purpose of AL is to score higher accuracy with few training labels as long as it has the ability to choose the data from which it learns[103].

In practice it makes sense to use SSL and AL in conjunction to cope with the challenge of scarcity of labeled data in different domains, and minimize the cost of data annotation by exploiting the abundant unlabeled data.

This chapter is devoted to the description of machine learning techniques, focusing on Semi-Supervised Learning and its extension Active learning, along to other algorithms we will be using in the next chapter for the implementation of our maintenance strategy.

3.2 Supervised Learning

The availability of annotated training data is a defining feature of supervised learning. The term «supervised»refers to the idea of having a supervisor instructing the learning system, on what label to associate to each training example. Supervised learning helps to optimize the performance criteria using past experiences, models are induced from the training data, and then used to classify other unlabeled data. This is done by learning a map between a given set of input X and an output Y, then using this map to predict output of unlabeled data that was not seen during the learning phase.

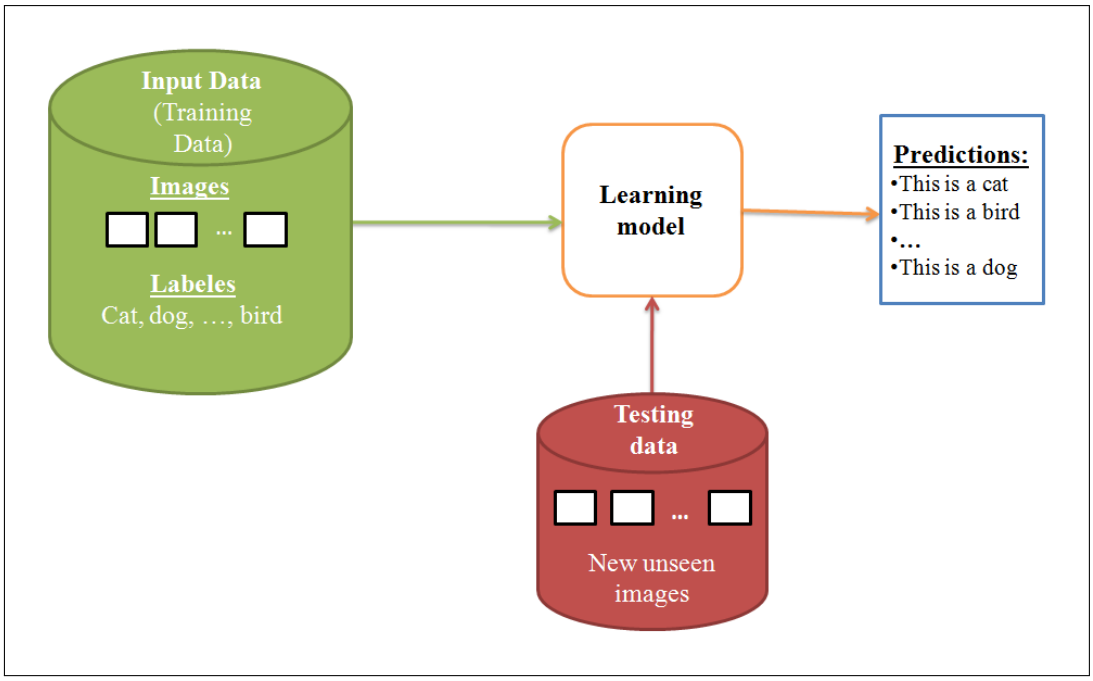

Figure 3.1 – *Supervised Learning[14].*

We can formulate Supervised Learning as follows[104]: Given an instance x_i a specific object, it is typically represented by features vector with a D dimension, Y a class labels, and K the number of output variables that an input object can have.

$$\text{X}= \{x_i\}_{i=1}^{N} \text{ and D } \in \text{R}^{D},$$

$$\text{X}= \{y_k\}_{k=1}^{K} \text{ presents the class labels of the } i_{th} \text{ object.}$$

Supervised learning problems are grouped into classification and regression problems, Classification is a method where an object is assigned a label(disease/no disease, red/blue/green), while regression is about predicting a continuous quantity output(a real value such as weight or dollars).

- If Y is a discrete value, it is classification;
- If Y is a continuous value, it is called regression.

Example: What is the temperature going to be today?

- Classification: Hot/Cold,
- Regression: 32°.

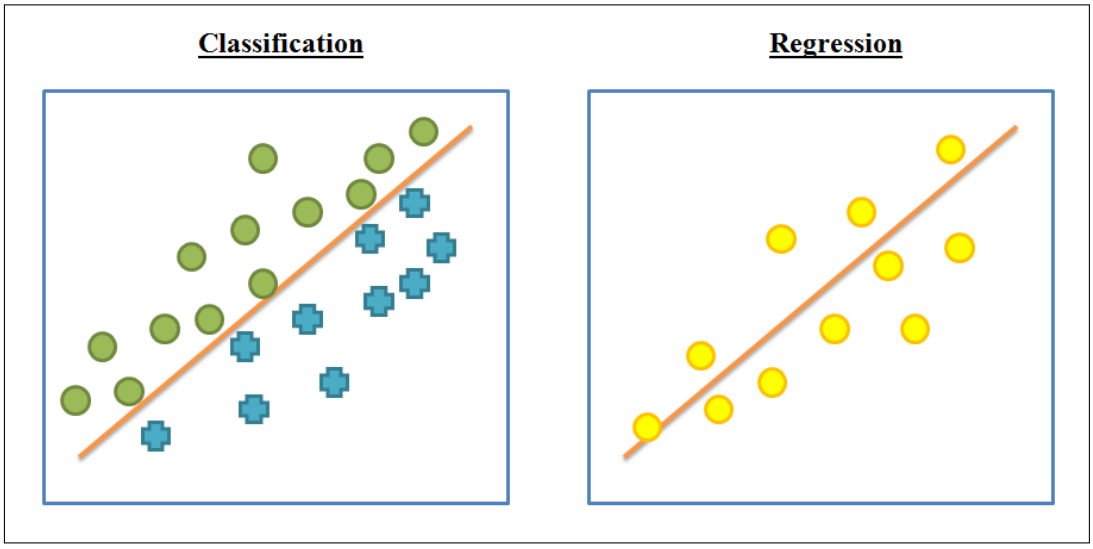

Figure 3.2 – *Classification VS Regression.*

Among classification algorithms we can mention: Naive Bayes, Support Vector Machine, Bayesian Network classifier[105].

3.3 Unsupervised Learning

Another technique of learning is unsupervised learning, known to be closer to true artificial intelligence [106]. Given a features vector $\{x_i\}_{i=1}{}^{n}$ and a similarity measure between pairs of vectors $K : XX \rightarrow R$, the goal of unsupervised learning is to partition the set, so that objects within each group are most similar to each other than the objects between the groups[104]. Figure3.3 show the unsupervised learning process. In unsupervised learning, the supervision of data is

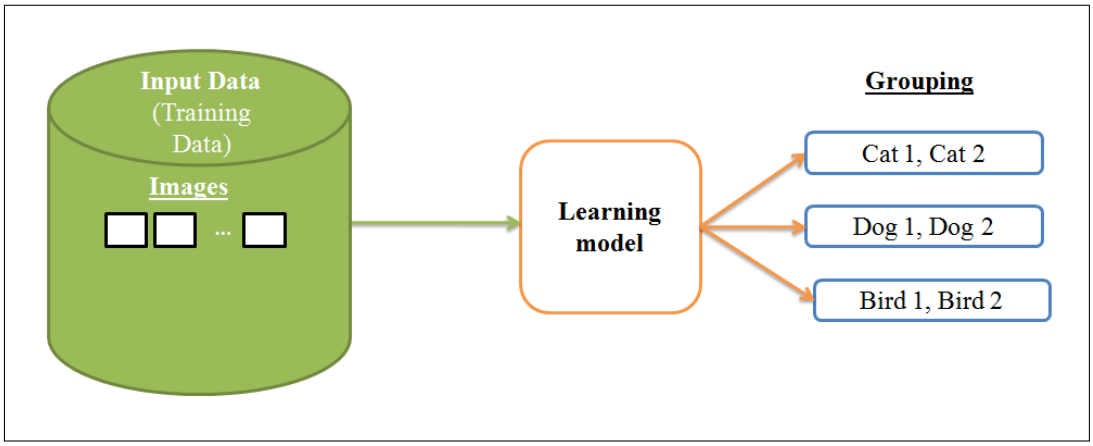

Figure 3.3 – *Unsupervised Learning [14].*

not required, instead, the model is enable to discover information while processing the unlabeled data. In this type of learning the right output is unpredictable, alternatively, the model explores the data and infer a function to describe a hidden structure or patterns from this uncategorized data. Some of the prime reasons to use unsupervised learning are:

- Ability to discover all kind of patterns in data,
- Able to find features that can be useful for categorization,
- Uses unlabeled data which is easier to get than labeled data that requires manual annotation.

Unsupervised Learning problems can be grouped into clustering and association problems:

1. *Clustering:* considered as fundamental data mining task, clustering is the process of partitioning data, based on a similarity measure into meaningful subclasses also known as clusters. Some of the models belonging to this family of unsupervised earning are: K-means algorithm and Fuzzy-K-Means (FCM) which is a version of K means (both algorithms will further be used in our study).

2. *Association:* an association rule learning problem aims at discovering rules that describe a large amount of data. It is about discovering relationships between data points in a data set (i.e., people that buy X tend to buy Y).

3.4 Semi-Supervised Learning

For the purpose of compromising and combining the power of both supervised and unsupervised learning semi-supervised learning (SSL) and active learning (AL) have been designed[107]. SSL refers to algorithms in which a combination of labeled and unlabeled data are used for the training of the model, in an effort to cope with the scarcity of labeled data, where annotation is costly and time consuming. Using both labeled and Unlabeled data for training is useful for the following reasons:

1. Annotation of a massive amounts of data for the supervised learning is as we mentioned can be prohibitively expensive and time consuming. Furthermore, excessive labeling can impose a biases on the model.

2. Including unlabeled data during the training process contributes to improving the accuracy of the model[18].

In this type of learning, training data is supplemented with a set of unlabeled data, the data set can be structured as follows[104] : $\{(x_1, y_1), ..., (x_L, y_L), x_L + 1, x_L + 2, ..., x_L + U\}$, $N = L + U$ where N is the size of the data set. Usually there is much less labeled then unlabeled data, which means: $L \ll U$. We define two slightly different scenarios of SSL, namely, *inductive learning* and *transductive learning*[107] (Figure3.4):

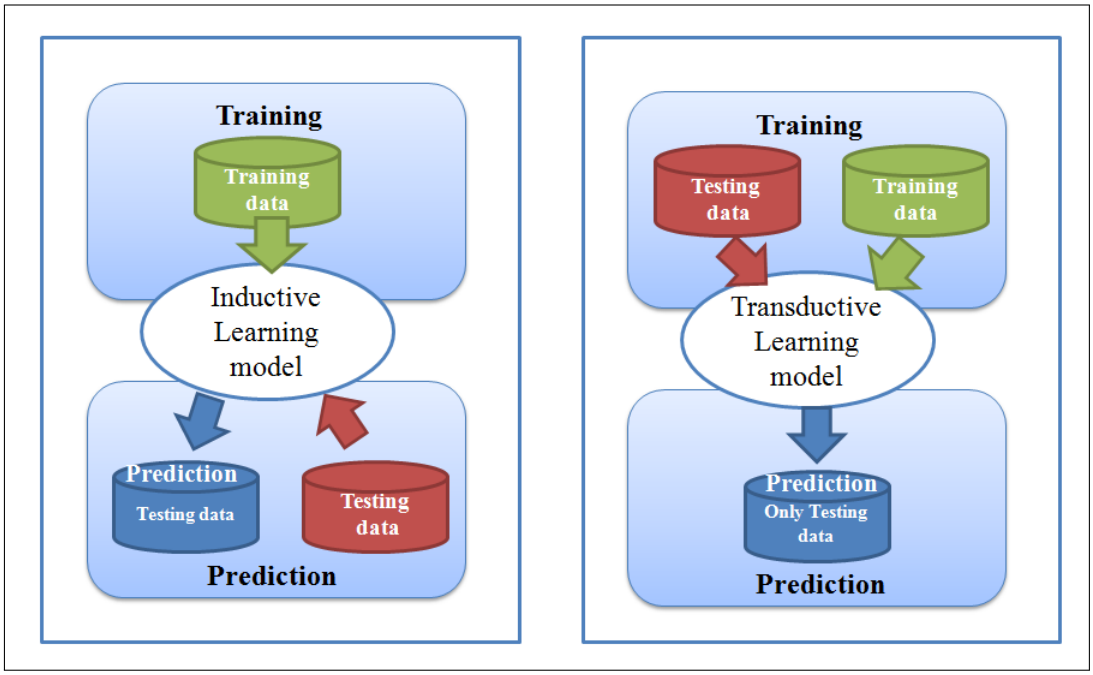

Figure 3.4 – *Inductive and Transductive Learning[15].*

1. **Inductive learning** is the commonly supervised learning approach, a machine learning model is built and trained on a set of labeled training set, then we use the trained model to predict labels of a set of tests never encountered before.

   Given the training set $\{(x_i, y_i)\}\ \{(x_i, y_j)\}_{i=1}^{l}, X = \{x_j\}_{j=l+1}^{l+u}$

   Inductive SSL builds a function $f : x \to y$ so that f is a good predictor of data going beyond $X = \{x_j\}_{j=l+1}^{l+u}$.

2. **Transductive learning** learning in this type of learning, the machine learning model has observed beforehand both the training and testing data. The model learns from the previously observed training set and then predicts labels for the testing set. Even if the labels of the testing sets are unavailable,

we can benefit from the patterns and any additional information present in the learning process. Given the training set ${\{(x_i, y_i)\}_{i=1}}^{l}$, $X = {\{x_j\}_{j=l+1}}^{l+u}$ inductive SSL builds a function $f : x^{l+u} \rightarrow y^{l+u}$, so that f is a good predictor of only unlabeled data that it has encountered during the training phase, it is not required to make external predictions.

The main difference between inductive and transductive learning is that during the latter, the machine learning model has already encountered both the training and testing set when training the model, and then uses the learned model to pedict only the unlabeled data points it has already encountered at the training phase. However, in transductive learning the model is trained only on the training set and then applies the learned model to predict labels of data points never-before encountered.

In the following subsections, we present fundamental semi-supervised learning methods [107]. The subsections correspond to different techniques, namely:*Self-learning, Co-learning, Graph-based methods and Transductive Support Vector Machine (TSVM)/ Semi-Supervised Support Vector Machine (S3VM).*

3.4.1 Self-training

A technique commonly used[108], it is perhaps the simplest and easiest semi-supervised learning technique to apply. This technique is characterized by the fact that, the learning process uses its own predictions to learn. Self-training is an incremental algorithm where the main idea is to first construct a function f on labeled data. The function f is then used to predict the labels of unlabeled data. A subset S of unlabeled data, together with their predicted labels, is then selected to increase the size of the labeled set[109].

The main idea of this technique is defined in these steps: Train-Predict-Retrain (using best prediction)-Repeat.

Algorithm 1 Pseudo code of Self-training algorithm

Given labeled data X_l $\{(x_1, y_1)\}_{i=1}^{l}$ and unlabeled data X_u $\{(x_1)\}_{i=l+1}^{l+u}$

Initialization:L=$\{(x_1, y_1)\}_{i=1}^{l}$,U= $\{(x_1)\}_{i=l+1}^{l+u}$;

1: **repeat**
2: Train f using L;
3: Query f on U ;
4: Remove subset S from U;
5: $L = L \cup S$;
6: **until** stopping criterion is met

3.4.2 Co-training

Co-training is similar to self-training with one critical difference. In self-training, one classifier is used to make predictions on unlabeled data, and then this data is fed back into the algorithm with predicted labels. Whereas in co-learning, two classifiers are used, each operating on a different view of the same instance. Assuming that features can be divided into two views(sub-feature set), and each sub-feature set is sufficient for the training of a classifier [110]. Each classifier predicts the unlabeled data and with the few labels it predicted "teaches" the other classifier.

Let f_1 be a classifier with view1, although we give it the full features vector x, it is only interested in the first view x^1 and ignores the second view x^2 (view2), f_2 is the reverse. They each provide their most confident predictions of unlabeled data as training data for the other view[104].

Algorithm 2 Pseudo code of Co-training algorithm

Given labeled data X_l $\{(x_1, y_1)\}_{i=1}^{l}$ (each instance has two views x_1, x_2), and unlabeled data X_u $\{(x_1)\}_{i=l+1}^{l+u}$

Initialization:L=$\{(x_1, y_1)\}_{i=1}^{l}$,U= $\{(x_1)\}_{i=l+1}^{l+u}$

Create a pool U' of examples by choosing u examples randomly from U;

1: **repeat**
2: Use L to train two classifiersh_1 that considers x_1 view of x , h_2 that considers x_2 view of x;
3: Select from U'_p most confidently labeled by h_1 as positive examples ;
4: Select from U'_n most confidently labeled by h_1 as negative examples ;
5: Select from U'_p most confidently labeled by h_2 as positive examples ;
6: Select from U'_n most confidently labeled by h_2 as negative examples ;
7: Add these self-labeled examples to L;
8: Remove them from unlabeled data U ;
9: **until** stopping criterion is met

3.4.3 Transductive SVM (TSVM)

A method also known as: Semi-Supervised Support Vector Machines (S3VM), it is an extension of standard SVM with unlabeled data.

TSVM take into consideration that the training set is split into two disjoint sub-sets[111], labeled data L, and unlabeled data U. TSVM algorithm aim is to exploit the unlabeled data in order to adjust the decision boundary that has been initially set from a small amount of L data[112] while going through the low density regions, it tries to keep the labeled examples correctly classified[107].

3.4.4 Graph-Based

A method where a graph is build from the training data.. Labeled data and weighted edges indicate the similarity, while nodes are unlabeled instances, and label information of each sample is forward to its neighbors[113]. Most of the graph-based Semi- Supervised learning mainly focus on how to conduct semi-supervised learning over a graph, and that will influence the learning performance is the form the graph is constructed with, that will seriously reflect the fundamental similarities among examples.

Each Semi-Supervised learning technique has its own advantages, we have gathered and displayed the main advantages of each technique (Table3.1) .

Table 3.1 – *Advantages of each SSL methods[18].*

SSL Method	Pros
Generative model	• Simple method
Self-training model	• Few labeled samples needed, • Classifier teaches itself using its own prediction.
Co-training model	• Interact with classifier, • Features evaluated simultaneously and consider dependency between them
Transductive SVM	• Effectively handles few labeled samples · • Compared to self-training and co-training the computational cost is less, • Interact with classifiers
Graph-based model	• Efficient method, · • Simplicity of computations, · • Better generalization ability.

3.5 Active Learning

A multitude of algorithms and applications for learning with queries have emerged in recent years, in this chapter will attempt to disseminate the basic ideas and methods that have been considered by the machine learning community. The driving idea behind active learning is that a machine learning algorithm can achieve better results with less training if it is allowed to choose the data from which it learns. An active learner may submit «queries », usually in the form of unlabelled data instances to be labelled by an "oracle" (e.g. a human annotator)that already has an idea of the nature of the problem[103].

In the following section, a general review of the literature on active learning is presented.

3.5.1 Definition

As outlined in the previous section, semi-supervised learning has attempted to address the problems associated with the need for labelled data to train a model, which in most cases is quite expensive in terms of time or computation. SSL minimises the cost of obtaining labeled data by using both labeled and unlabeled data during training. The problem encountered in this type of training is the use of the entire unlabeled data set without prior selection of the data that provides the most utility to the classifier.

An alternative solution would be to choose to label only a subset of the available data, but the choice of the subset will affect the quality and performance of the final model. The question is therefore: *How to select the subset of data that will give the best performance of the model? performance of the model?*

The importance of Active Learning is emerging in applications that deal with large amounts of data. Active Learning is most suitable when there are numerous unlabelled data instances, they can be easily collected or synthesised, and you expect to have to label a large number of them to form an accurate system. Since labeling such data can be very costly and exhausting. Active learning is an iterative machine learning algorithm in which the main problem is to evaluate the informativeness of an unlabeled instance[103](see Figure3.5).

The active learner is a classifier initially trained on a few labeled instances. Then, attractively, and based on its knowledge derived from the labeled data, it requests a label for one of the unlabeled data instances. Successful active learning should result in a significant reduction in the amount of training data without a significant reduction in classifier performance (Figure3.5).

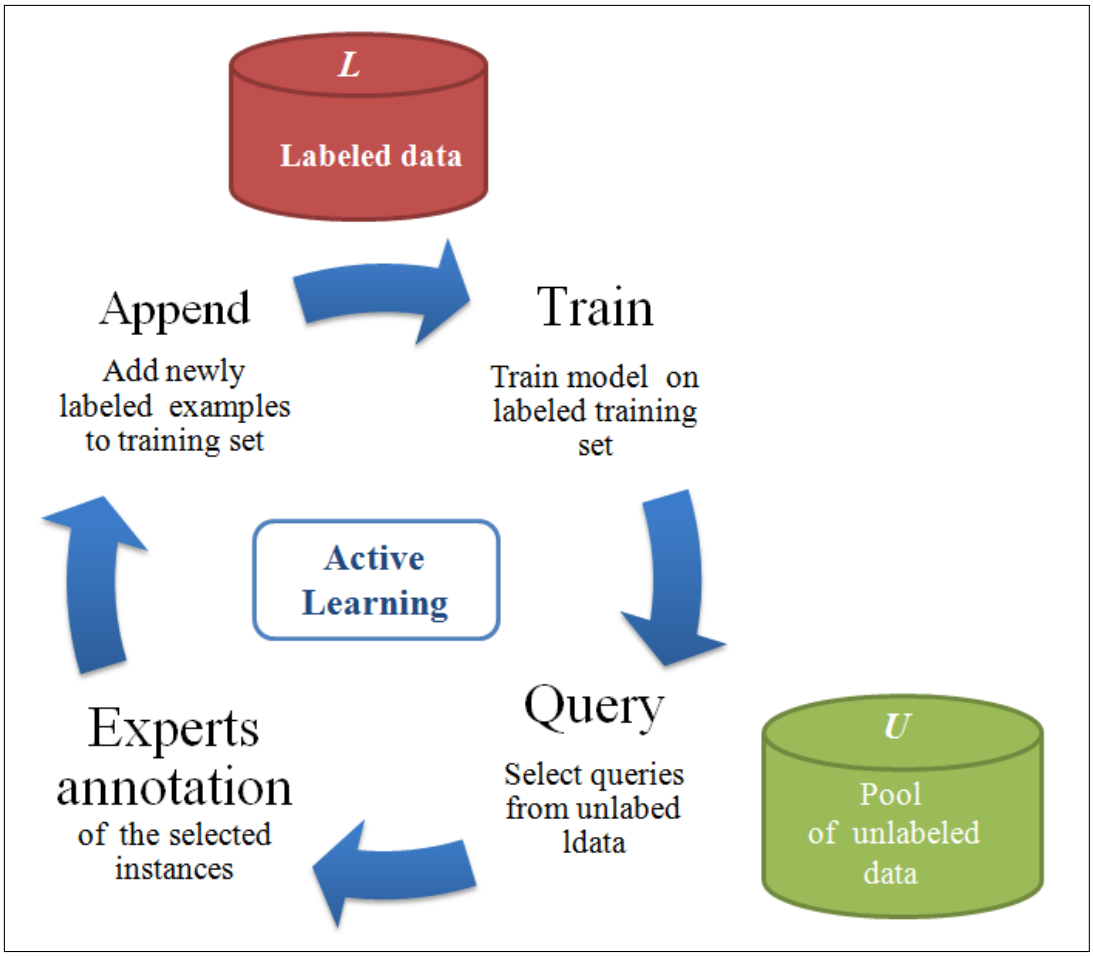

Figure 3.5 – *Active Learning process(Pool-based scenario).*

3.5.2 Active Learning Scenarios

There are different scenarios in which the learner can request queries. The three main settings that have been considered in the literature are: ***Query synthesis***, ***Stream-based selective sampling***, ***Pool-based sampling***[103].

3.5.2.1 Query Synthesis:

One of the first active learning scenarios to be investigated is learning with Membership Query[114]. In this type of scenarios, the learner can request labels for any unlabeled instance in the search space. Query synthesis is often effective for classification domains[114]. The idea of query synthesis has also been extended to regression learning tasks, such as learning to predict absolute coordinates of a robot hand given the joint angles of its mechanical arm as inputs[115].

3.5.2.2 Stream-based selective sampling:

Cohn and his colleagues introduced Stream-based Active Learning, in which the key idea is to select one instance at a time from a sequence of instances [116][117]. Then the active learner, based on the information measure of the instance, must decide whether to query this instance or to ignore it. Stream-based Active learning is typically used when data cannot be easily stored. An advantage of this scenario over other active learning scenarios is the fast query decision making[104].

3.5.2.3 *Pool-based active learning:*

Pool-based Active Learning introduced by Lewis[118] is the most common approach in learning and data mining applications. In this approach, instead of sampling one instance at a time a large number of instances are sampled and then the model selects the best query to label. As a result, there are a small number of labeled L and a large number of unlabeled U.

The main difference between active stream-based active learning and pool-based active learning is that the former analyzes the data sequentially and makes query decisions individually, while the latter evaluates and ranks the whole set before selecting the best the whole set before selecting the best query.

3.5.3 Sampling criteria

Active learning is an iterative sampling+labeling procedure, sampling is the he process of selecting data points to be labeled. The objective of active learning is to find at each iteration, the most useful and valuable instances among a group of unlabeled data. These instances are used to relearn the model and expected to improve its performance.

There are three main sampling criteria for an effective active learning algorithm, that is: informativeness, representativeness and diversity[119],[120]:

1. *Informativeness:* aims at selecting unlabeled data to add rich information to the current classifier,
2. *Representativeness:* aims at selecting unlabeled data with high representation, it means the samples with high density, so that it represents more neighbouring,
3. *Diversity:* aims at selecting samples that scatter the entire input space, instead of focusing on one small region of it.

3.6 Semi-Supervised Learning in medicine

Our research question emerged when we questioned the issue of coping with the scarcity of labeled data, particularly for medical applications. Semi-supervised learning techniques have been considered by numerous researchers in the past few years, and we were interested to explore how this type of learning would perform when applied to medical data for the implementation of a CAD system.

We presented a survey paper entitled «Improving the performance of computer-aided diagnosis systems using semi-supervised learning: a survey and analysis»[18] to evaluate the to see the impact that this type of learning can have when in contact with data as sensitive as medical data.

Given the rapid growth in the size of the data generated in the health care sector, computer-aided diagnosis has become an essential part of health management. In the field of health care, computer-aided diagnosis has become an essential tool for most medical experts to help them make decisions. These systems are mostly determined by the large volume of supervised (labeled) data sets required for their implementation, and need to be labeled by human experts as mentioned before. A selection of different CAD systems that adopt a semi-supervised learning approach presented in Table3.2. The Table displays the different SSL techniques used for different tasks, presented in literature between $[2009 - 2020]$.

Table 3.2 – *SSL techniques used for medical applications*

Techniques used		Year,Ref	Application Domain (Study)
Self- training	/	2019,[121]	Glaucoma detection
		2017,[122]	Cardiac segmentation
		2017,[66]	Biomedical image segmentation : Vessel and Neuron Segmentation
Co-training	/	2020,[123]	NIH Pancreas + LiTS liver tumor
		2016,[124]	Automated identificationof lung sounds
		2016,[125]	Mass classification
		2009,[126]	Pulmonary nodules detection
	Expending of Co-Training: Co-Forest	2018,[127]	Automatic glaucoma screening
		2013,[128]	Classification of large biological data
		2010,[129]	Pulmonary nodules detection
		2007,[130]	Classification+ Microcalcification detection in digital mammograms
	Random Forest based SSL (RF-SSL) +Active Learning	2016,[131]	Segmentation of Crohns disease

S3VM (TSVM)	/	2016,[132]	Classification of mammographic images
		2015,[133]	Classification of Mild Cognitive Impairment
		201,[134]	Classification of breast cancer
		2010,[135]	Classification of Mild Cognitive Impairment
Graph-Based	/	2017,[136]	Aneurysm volume estimation
		2017,[137]	Classification of neurodegenerative disease
		2016,[138]	Liver segmentation
		2016,[139]	Classification of Alzheimer's Disease
		2016,[140]	CBIR Prostate for the Classification of glands
		2016,[141]	Breast cancer diagnosis
		2015,[79]	Detection and diagnosis of suspicious regions in the mammograms
		2014,[142]	Segmentation in four applications
		2014,[143]	Prediction of Cancer (Colorectal Cancer and Breast Cancer)

Graph-Based	Graph -based +Active Learning	2014,[144]	prostate segmentation
	/	2014,[145]	Classification of Breast Cancer on UltraSound Images
		2012,[146]	The Automatic Wall Motion Abnormality Detection
		2011,[147]	Classification of Alzheimer's Disease and Mild Cognitive Impairment
		2011,[148]	Classification of Breast Tissue Breast cancer classification
		2011,[149]	Identifying potential disease-miRNA association
		2009,[150]	Diabetes disease prediction
		2008,[151]	Segmentation of nasopharyngeal carcinoma lesion

Combined Semi-Supervised Learning techniques	-Self- training -Graph- based	2016,[152]	Optic disc missing annotation prediction
	-Self- Training + Co- Training	2014,[153]	Automated detection of Microaneurysms
	-Self- Training -Co- Training + Active Learning	2010,[154]	Classification of tuberculosis patterns in CT
Novel Semi-Supervised Learning proposed methods	Federated SSL	2020,[55]	COVID 19 detection using CT images
	Self-ensembeling model	2018,[155]	Automatic skin lesion segmentation
	Cluster- then- labeled	2018,[156]	Triaging breast digital pathology image patches and classification of nuclei figures
	A proposed Semi-supervised learning method applied to train a neural network model	2018,[157]	Cardiac pathology classification
	Self-advised SVM	2017,[158]	Diagnosis of Skin Cancer

The objective of this survey was to define the influence of SSL methods in the medical field, used for the development of CAD systems due to the scarcity of annotated data. The results obtained by many research papers in the literature have shown the effectiveness of exploiting unlabelled data when combined with the limited labelled data available. Some of the key findings include :

- Active learning used in conjunction with semi-supervised learning decreases the cost of annotation, by letting a learning algorithm choose the unlabelled samples to label,
- Taking advantage of a few annotated samples and abundant unlabelled data is very promising and has a significant impact on the application of CAD systems,
- For segmentation tasks, the introduction of unlabeled data leads to improved segmentation performance, especially when the size of the existing training set is small,
- Semi-supervised learning compared to supervised learning improves the overall system performance when both use the same number of labelled samples,
- Mainly, unlabeled data improves classification performance, when the assumed model is correct. The enhancement of the performance is highly dependent on the number of labeled data, their correct and accurate annotation, and the complexity of the problem.

3.7 Ensemble Learning

One of the most significant goals of machine learning is to build learning systems with a strong generalization ability. Zhou[159] demonstrated that both semi-supervised learning and ensemble learning are machine learning algorithms beneficial for each other. The former aims to reach a strong generalization by exploiting unlabelled data, and the latter attempts to reach a strong generalization by using multiple learners. It is worth noting that even the two paradigms have been recently applied in conjunction for numerous real-world problems, the were developed separately.

Ensemble learning has been used successfully in machine learning in differed application namely, computer-aided diagnosis, speech recognition, text categorization. Assigning labels to data points described by a set of measurements is what Pattern recognition field is all about; the purpose is to discover a structure within the data set, and to be able to identify what are the characteristics (features or attributes) that make certain points similar inside a group and different across other groups. Pattern recognition is closely linked to Machine Learning; field where algorithms are used to learn from the data and make predictions.

Combining classifiers is a colossally growing research area, getting major attention from communities such as Machine Learning and Pattern Recognition, this learning algorithm is also known as Ensemble Learning, where a set of classifiers is constructed to classify new data points, their individual decisions are combined using different techniques (Typically weighted or unweighted voting)[160] to drive a consensus decision. in other words, what is meant by an ensemble, is a set of individual predictors, of which the predictions are combined for a given classification task[16](see Figure 3.6).

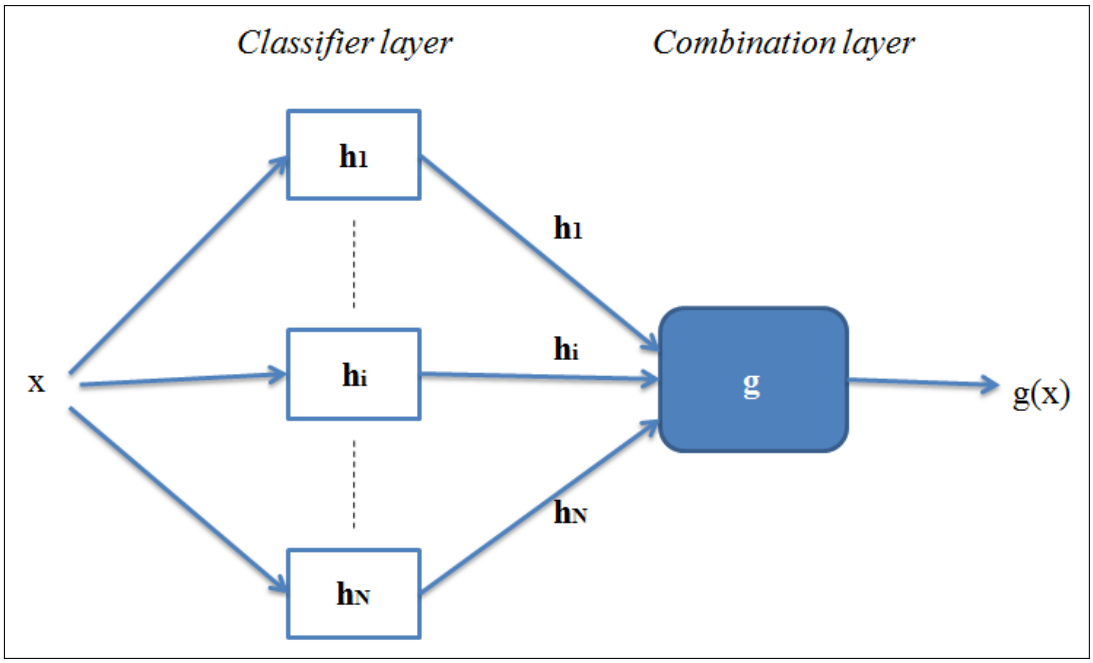

Figure 3.6 – *Two layer architecture of an ensemble [16].*

The purpose of combining several classifiers that adopt algorithms with different strengths and weaknesses is to provide a system that can achieve greater accuracy and outperform the individual algorithms within the system itself[161]. Appearing under a variety of notions in literature: Ensemble Learning, Multiple Classifier System (MCS), Classifiers Fusion, Divide-and-Conquer Classifiers. Essential concepts of ensemble learning are presented in the following classic literature[162].

Dietterich presented reasons why it is advantageous to use an ensemble learning method [160]. These reasons are the main drawbacks of existing basic learning algorithms, which ensemble learning seeks to minimise or eliminate. Among these reasons, we can distinguish a particular problem that we wish to address in our study namely «**The statistical problem** ».

- ***The statistical problem:*** is a problem that occurs when we have a limited training data set. Given a hypothesis space F (the space of all possible classifiers), a base learning algorithm *BaseLearner* searches the space F to identify the best single classifier f. When disposing of a small training data compared to the size of the classifier space, *BaseLearner* can not identify f. Even though

the data is limited, the *BaseLearner* can still find numerous classifiers within F that give good accuracy on the training data. Building an ensemble H of these accurate classifiers leads to a good approximation of f.

An important point that needs to be highlighted is the choice of algorithms to build the ensemble, different algorithms can provide different structures of the same data set, therefore the only indication for determining the quality of the results is a subjective estimation of the user[162].

3.7.1 Diversity

Dietterich[160] stated that it is possible to construct an ensemble of classifiers that is more accurate then one single classifier. Yet, for an ensemble of classifiers to be more accurate than the individual classifiers that compose it, a fundamental condition must be met, which is: «Diversity »[160].

We refer to diversity among classifiers as the independency of errors(uncorrelated) which means that they have distinct missclassified examples. The following example clarifies the need of diversity when constructing an ensemble of classifiers: we have a set of three classifiers $\{h_1, h_2, h_3\}$, considering a new case x ,and the ensemble of three identical classifiers means that if h_1 (x) is wrong , automatically h_2 (x) and h_3 (x) are also wrong, meanwhile if errors made by classifiers were uncorrelated , then if h_1 (x) is wrong , it is possible for h_2 (x) and h_3 (x) to be correct , in this case the majority voting will correctly classify x.

3.7.1.1 How to Create Diversity?

To group a set of classifiers, one must first find a method to build the desired ensemble. Figure3.7 provides a graphical illustration of the different approaches presented by Kuncheva [162]to built diverse combination of classifiers.

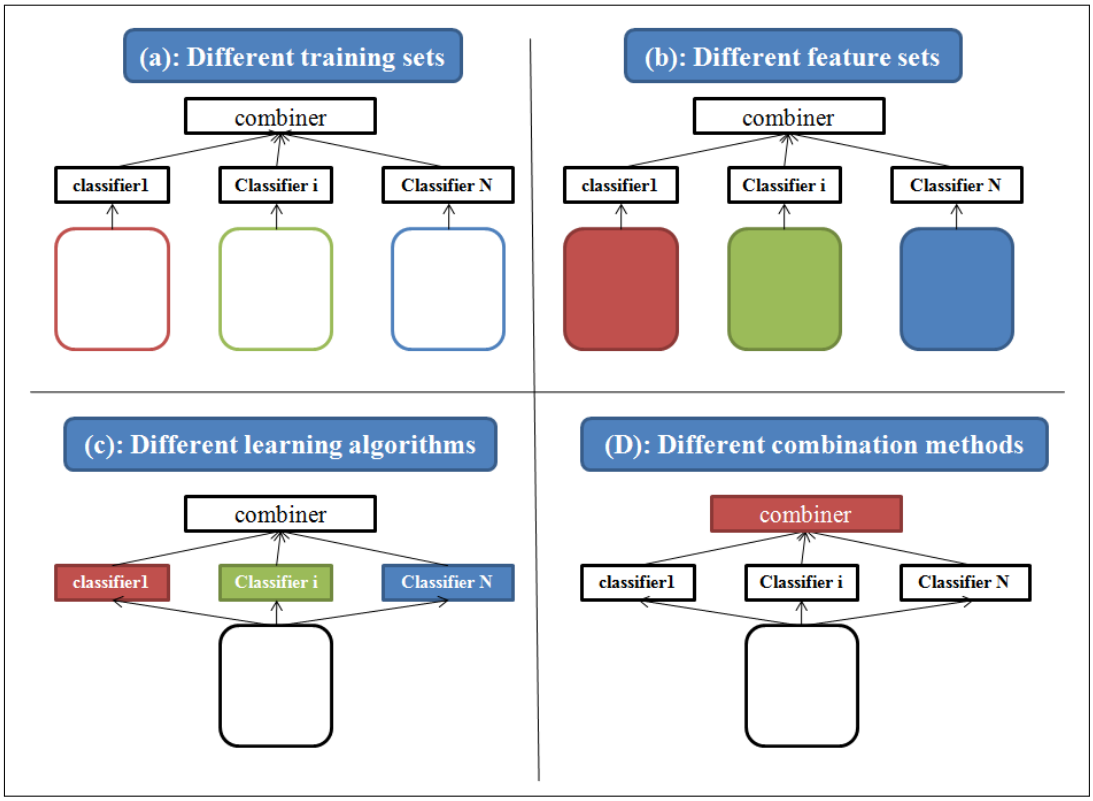

Figure 3.7 – *Four approaches to create diversity among classifiers [16].*

As mentioned earlier, the technique used to promote diversity among the member classifiers of an example is what makes a distinction between the different ensemble methods(Figure3.7):

1. **Approach (a):** targets the *Data level*, where different training sets are manipulated ,and each classifier in the ensemble is trained on its own data set.This particular approach once put to test has proven to be very successful, considering the use of bagging and boosting methods,
2. **Approach (b):** targets the *Feature level*, where different feature sub-sets are used to train each classifier from the ensemble,

3. **Approach(c):** targets the *Classifiers level*, where a heterogeneous ensemble method is adopted to combine classifiers trained on different learning algorithms, Contrarily to homogeneous methods, where the set of classifiers is trained using the same learning algorithm. Various ensemble paradigm tend to employ the same classification model, but there is no clear evidence that this technique is exceptional compared to using different models[162],

4. **Approach (c):** targets the *Combination level*, where different techniques to combine the classifiers decisions are used, the main task at this level is to select the best combination method, assuming that we have a set of given divers classifiers.

3.7.2 Ensemble learning Algorithm

In the following subsection, we present the ensemble method that is used our this thesis. To build a diverse combination of classifiers as explained in section3.7.1.1, we chose to operate on two different levels: the **Data level** *(approach a)* and the **Classifiers level** *(approach c)*.

3.7.2.1 Manipulation of the training set: Bagging

In this work we try to cope with the scarcity of labeled data that compose the training set. Given that the labeling process is considered a challenging and expensive tasks that demands a lot of time and effort from expert human annotators.

The Bagging algorithm allows us to train our set of classifiers with different subsets (samples) that are contained in the existing small training set, a small change in the training set can lead to a remarkable change in the output of the classifiers. To manipulate the training set we use «*Bagging* » a method proposed

by Breian[163]. Bagging is a learning algorithm where the main idea consists of sampling the original training set L to m training examples, drawn randomly from L (Figure3.8). Such training set is referred to as Bootstrap aggregation, and each Bootstrap replicate includes 63.2% of the original training set L, with many training examples appearing numerous times.

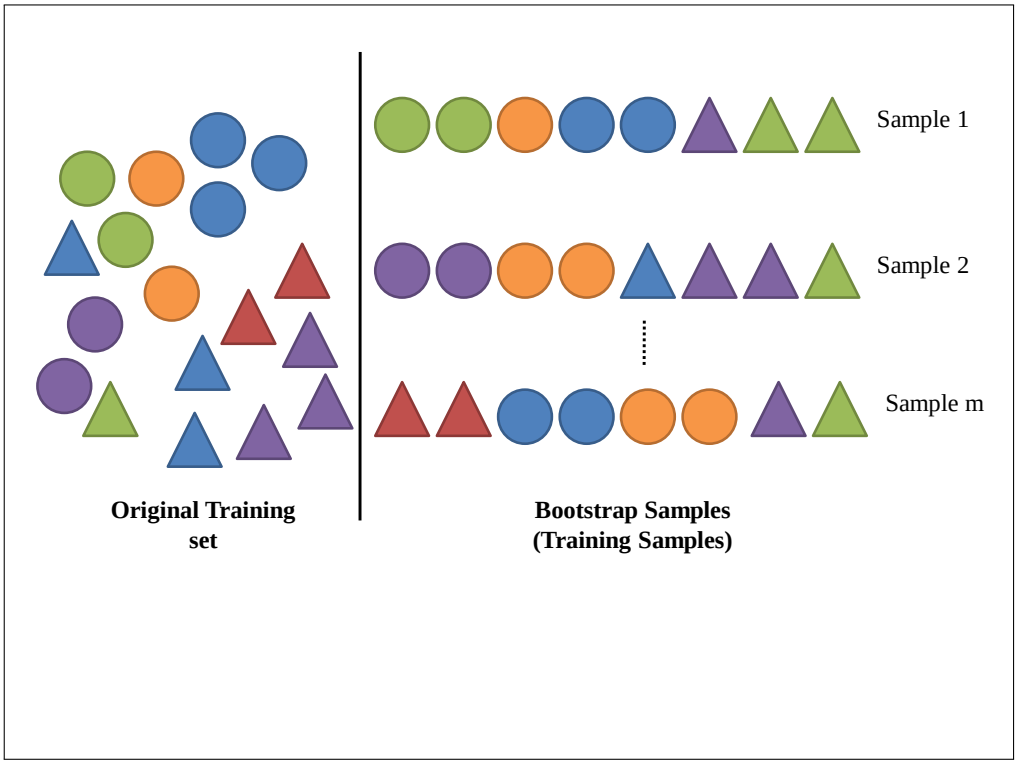

Figure 3.8 – *An example of bootstrap sampling.*

In Bagging the needed diversity for our ensemble is created by manipulating the generated training samples, which allows to create several hypothesis. The learning algorithm then is run multiple times, each round with a distinct bootstrap of the training samples(Figure 3.9).

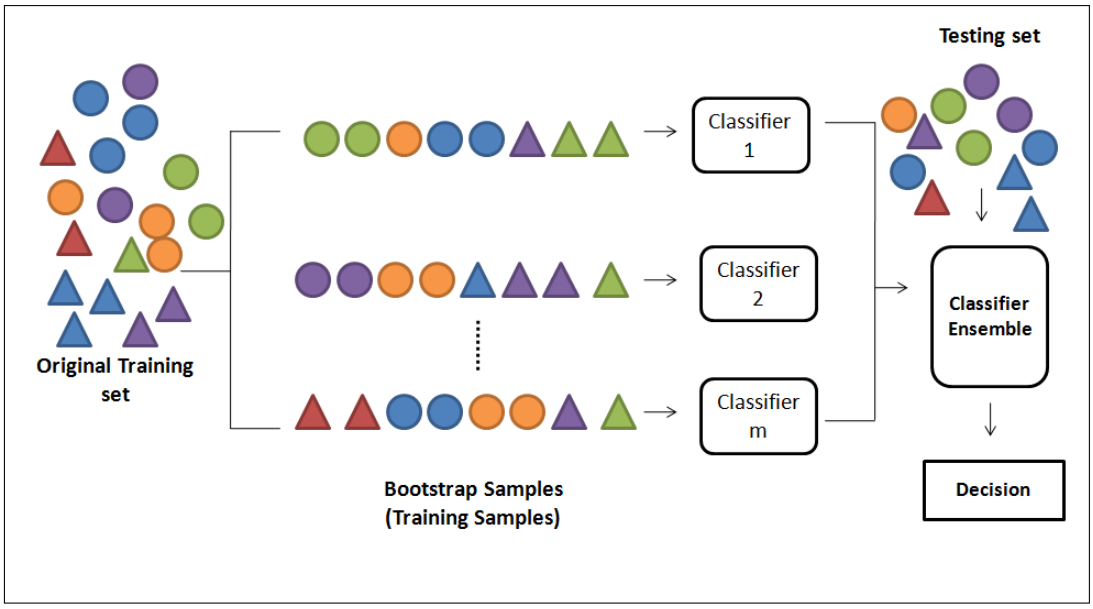

Figure 3.9 – *The Bagging algorithm.*

3.7.2.2 *Manipulation of the learning algorithms*

There are two distinct approaches for Combining Multiple Classifiers (CMC): **Classifier fusion** and **Dynamic classifier selection** [164].

1. Classifier fusion: in this approach, the individual classifiers are used in parallel and their outputs are combined in some manner, e.g. by a majority vote, to obtain a «group consensus ».

2. Dynamic classifier selection: This approach is intended to predict which single classifier is most likely to be correct for a given example. Only the output of the selected classifier is taken into account in the final decision.

As our goal is o create diversity, we choose the first approach to combine the classifiers, namely the classifier fusion. As illustrated in Figure 3.7, Kuncheva [162] presented different approaches to create diversity among an ensemble of classifiers. One of the proposed approaches creates diversity by operating at the classifiers level *(Approach**(c)**)*. This heterogeneous ensemble method (Figure 3.10) combines classifiers trained using different training algorithms[16].

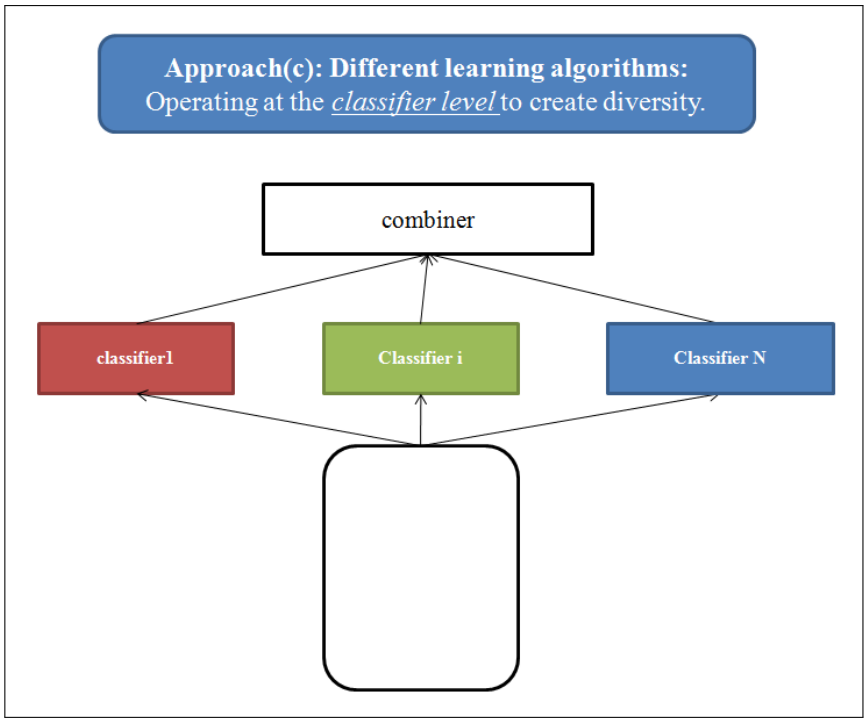

Figure 3.10 – *heterogeneous ensemble method[16].*

3.7.2.3 *Learning algorithms*

In this section, we briefly present the machine learning algorithms used to create our heterogeneous ensemble as discussed in the prior Section 3.7.2.2. We used the following algorithms for their known performance in terms of efficiency and classification.

The classifiers used for our ensemble learning are presented in the following subsections:

- **Bayesian Network**: in data analysis and pattern recognition, one of the most important and fundamental tasks is classification. This task involves the construction of a classifier whose function is to assign class labels to instances described by a set of attributes.

  One of the most effective and widely used classifiers nowadays is Bayesian Network, also called: belief network, decision network or Bayesian model.

Bayesian network a graphical model that represents a set of variables and their conditional dependencies via a direct acyclic graph (DAG)[165], it is a marked graph that represent the joint probability distribution. A Bayesian network consists of two parts:

1. Direct acyclic graph,
2. Table of conditional probabilities.

The graphical part of a Bayesian network reflects the structure of a problem, Figure3.11 illustrated a Bayesian network used for medical diagnosis[17].

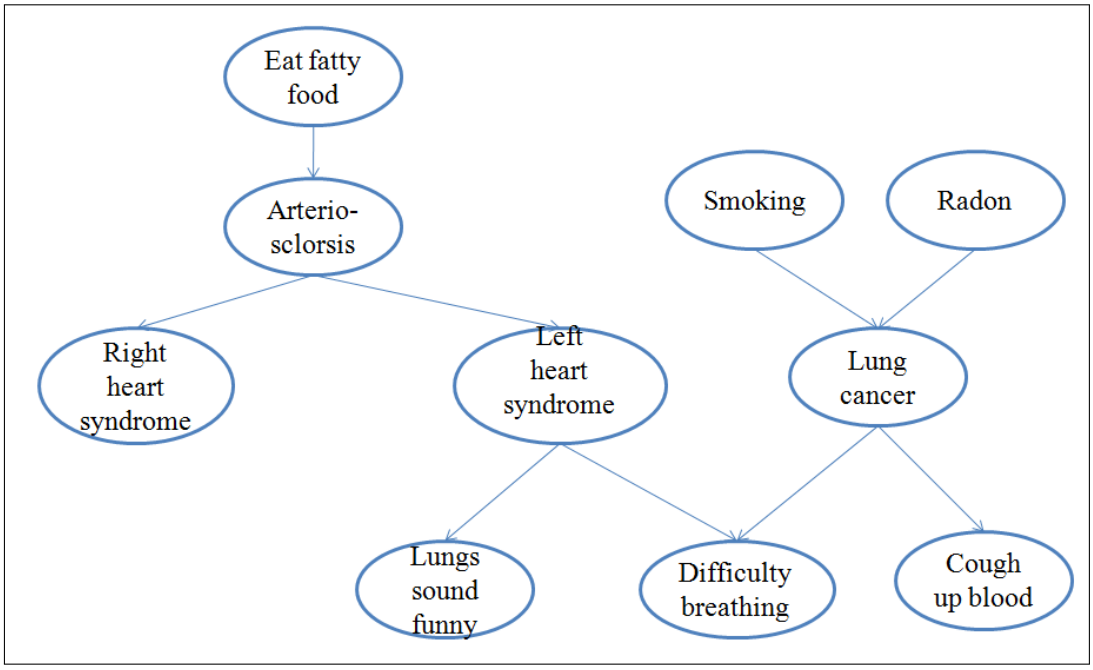

Figure 3.11 – *Bayesian network for medical diagnosis [17].*

- **Logistic Model Tree (LMT)**: tree induction methods and logistoc regression are two well-known techniques used for supervised learning tasks, for both prediction of nominal classes and numeric values.

  LMT is a classification model that was born out of the idea of brings together these two complementary and most popular classifications schemes, namely, tree induction and logistic regression[166]: a decision tree that has a linear regression at its leaves, the advantage is that estimated class probabilities are produced rather than just a classification. LMT is a more natural way to deal with classification tasks, it have been shown to be a very accurate classifier,

with a competitive performance compared to the state-of-the-art classifiers, while being very easy to interpret[166].

- **Support Vector Machine (SVM)**: one of the supervised machine learning algorithms, SVM provides an analysis of data used for classification tasks. An SVM classifier works as follow: the classifier searches for a hyperplane with maximum margin and support vectors for data during the training set. The generated hyperplane as well as the support vectors can be regarded as the decision boundary that separates the data points of one class from another[167](Figure 3.12). In Figure3.12 we can easily distinguish two categories, each identified by C_1 and C_2, this situation is a binary classification problem. SVM were developed for binary classification problems, eventually extension were made to allow this technique to support multi-class classification, for instance: Sequential minimal optimization (SMO) which is the simplest training procedure used to implement a multi-class SVM[168].

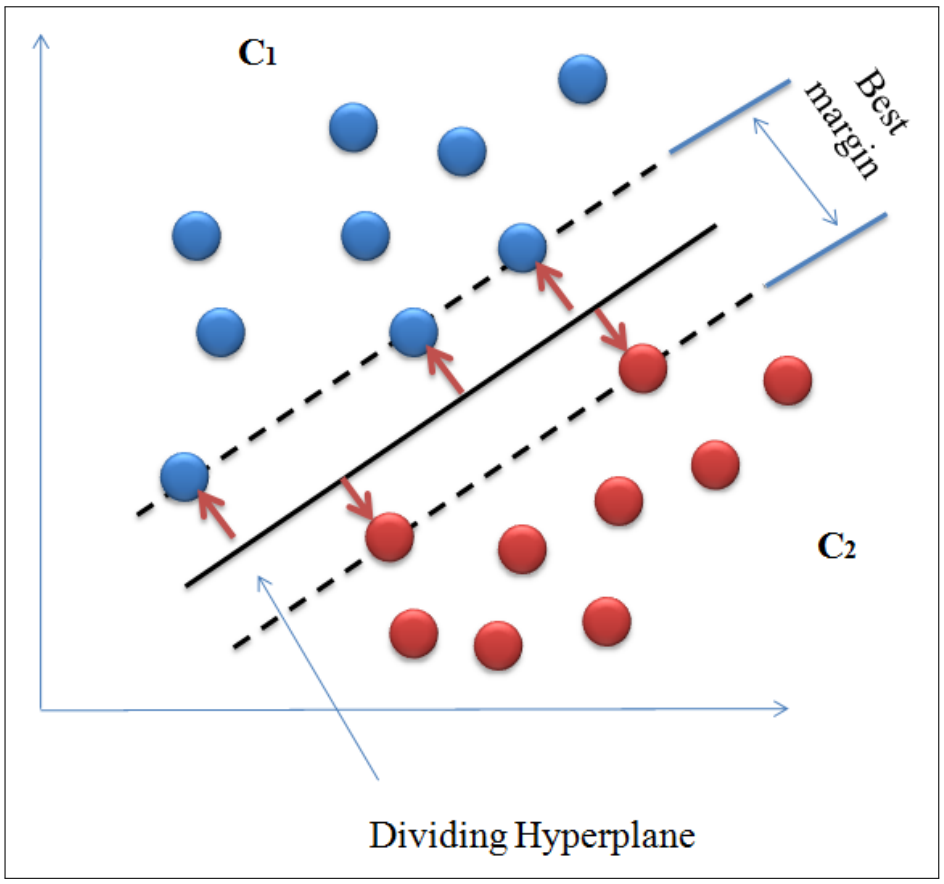

Figure 3.12 – *Data set representation and margin for SVM.*

These three supervised learning algorithms used as base learners for the ensemble learning are one of the most effective and popular machine learning algorithms for classification problems [105].

3.8 Conclusion

Machine learning has become the tool of choice for many applications. However, successful applications of machine learning often rely on the existence of large amounts of labeled data, which can be difficult to obtain. Over the past recent years, the research community has focused on semi-supervised and active learning to supervised and active learning to learn and benefit from information from unlabeled data.

This chapter has introduced machine learning techniques with a particular focus on on the semi-supervised and active learning techniques used in the following chapters. On one hand, semi-supervised learning has attracted a lot of interest, as it makes use of the large amount of readily available unlabeled data to improve classifier performance. Semi-supervised learning has been successfully applied in various pattern recognition applications.

On the other hand, active learning has received increasing interest in recent years by allowing machine learning researchers to choose among unlabeled instances those that will provide more information to the classifier.

Maintenance at the development stage: Active Semi-Supervised Maintenance (ASSM) approach

Contents

4.1 Introduction

As discussed in chapter 2, case base maintenance can be performed at different stages of the CBR life cycle. Maintenance is generally defined in software engineering and knowledge engineering as an activity that takes place after the development of the system is completed, and the application has already been deployed to exploitation. It draw our attention the fact that most CBR works focus on the life cycle of the system once it is operational, or on maintaining the knowledge containers to avoid performance degradation. Yet, to acquire a CBR system, it must first be developed. Little attention is paid to the development stage of CBR systems, or the problems that can be encountered while developing a case based reasoner.

In our work we consider the first two step of development which are data collection and case acquisition. We can easily face the problem of collecting data that must then be processed, refined and then structured to have the form of cases (Problem, Solution). This difficulty changes from one application domain to another. In order to have an initial CB that allows CBR system to be operational and capable of reasoning, labeled data are necessary. If we take the scenario of a CAD system implemented using CBR framework, we will quickly be confronted with the problem of scarcity of labeled data, that needs to be annotated by human experts.

For our contribution we take into consideration labeled, along with unlabeled data, where we propose an Active Semi Supervised Learning approach to build and maintain a quality CB at the development phase, as it is considered maintenance on a zero basis. The objective is to select the most "valuable" unlabeled data using an Active Learning sampling engine, then labeling the selected data using an inductive SSL model. The selected and labeled data points are then stored in CB to enhance and empower the reasoning performance of the system.

We can define active learning as an extension of semi-supervised learning that improves on it by adding the ability to select «valuable »data appropriately, which will significantly reduce the annotation cost. In recent years, active learning has gained a lot of interest because it aims to minimize the cost of labeling while maximizing the performance of the classifier. The key problem in active learning is to decide whether an instance is «valuable»or not. We propose two sampling strategies including three sampling criteria (informativeness, representativeness and diversity). The proposed maintenance approach is mainly based on SAMPLING, encompassing two fundamental points:

1. '*Sampling engine using Active Learning* to select most valuable data points from a pool of unlabeled data, using three sampling criteria (informativeness, representativeness and diversity);
2. *Learning engine using an inductive Semi-Supervised learning* to label the selected instances, and re-selected again the most informative sample to be retained in the CB.

In the following section we will present the proposed approach, as well as the details of each step. A discussion of the results obtained will follow, along with a comparison with some state-of-the art methods.

4.2 Proposed approach for Active Semi-Supervised Maintenance (ASSM) at the development stage of CBR

We often tend to consider CBR systems once they are operational, to discuss either the application phase, the maintenance phase or the different steps of the CBR life cycle. However we rarely pay attention to a primordial phase, that allows the actual implementation of a case-based reasoner, namely, the development phase.

To develop a CBR system the first step to consider is the acquisition of a case base, that enable the system to be operational and capable of reasoning. This task proves to be particularly challenging in some application domains i.e., when one wishes to implement a computer aided diagnosis (CAD) medical system using a CBR framework. The acquisition of a case base in such scenario requires expert human annotators (radiologist, doctors,...), this task is very costly in terms of time and effort and it is considered as a burden for human experts. On the other side, huge volumes of data sets are collected but most of them lack the supervised information [18].

To cope with the mentioned constraints, we propose a support tool to maintain while building a quality case base, which will have repercussions on the reasoning performance of the system, given that its quality is directly related to the quality of the case base.

The novelty of this approach lies in the integration of maintenance at the development stage, where machine learning techniques, namely semi-supervised learning and active learning are used to cope with the challenge of scarcity of labeled data, by taking advantage of the volumes of unlabeled data easily available. Semi supervised learning and active learning are used in conjunction in order to effectively select from the pool of unlabeled data the most valuable instances, this selection is done using three sampling criteria, and using the valuable selected instances along with the few available labeled data.

The objective is to build a competent case base able to perform a very good job at reasoning while reducing the annotation cost. We are interested in achieving both satisfying classification accuracy while monitoring the storage size of the case base, to avoid blind retention of cases. A detailed discussion of the different steps of the Active Semi Supervised Maintenance (ASSM) approach are details in the next subsections.

The proposed ASSM is composed of two phases (engines), the sampling phase where active learning is used, and learning phase where an inductive semi-supervised learning is used (see Figure4.1).

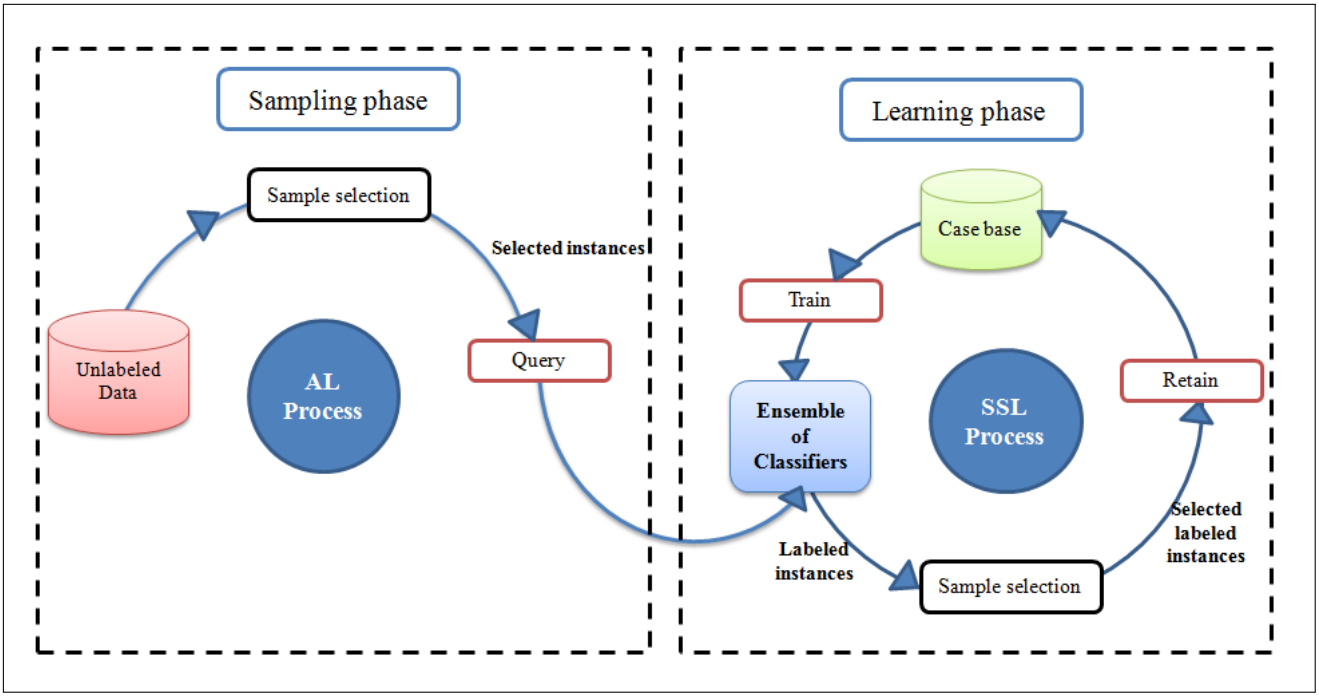

Figure 4.1 – *Architecture of the proposed ASSM with the Sampling phase and Learning phase.*

4.2.1 Sampling phase

The process of selecting data points to be labeled is called sampling. A key element of an active learning algorithm is the selective sampling strategy (selection strategy), as it allows the selection of instance to be labeled and added to the training set to improve classification performance.

Active learning (AL) is sometimes referred to as query learning, and since a supervised system requires for training at least hundreds (even thousands) of labeled instances to perform well, while for many supervised tasks labeled instances are very difficult to obtain and expensive. The key theory of active learning is that a learning algorithm is allowed to select the data from which it learns[103]. AL attempts to achieve high accuracy with few labeled data.

There are several scenarios for AL to operate and pose queries, additionally, there different sampling criteria that are used to select the most valuable sample. Pool-based AL is a well known active learning approach, in which queries are selected from a pool of unlabeled data D_U.

4.2.1.1 *Pool-based sampling*

Given a pool of unlabeled data , pool-based AL focuses on selecting most valuable instance using sampling criteria, so that once labeled, the model built from these instances can achieve the best possible performance[169].

Problem definition Given a small initial labeled data set $D_L = \{(x_1, y_1), ..., (x_M, y_M)\}$ (represents the case base of the CBR system), and a large pool of unlabeled data $D_U = \{x_1, ..., x_N\}$, each instance $x_i \in R^d$ a d-dimensional feature vector. $y_i \in \{0, 1\}$ is the class labeled of xi in the case of binary classification, or $y_i \in \{0, ..., k\}$ for multi-class classification.

At each iteration AL selects a batch of a size S from the pool of unlabeled data D_U, and queries their label from an oracle (in our case it is our ensemble of classifiers considered as experts). D_L and D_U are then updated, and the ensemble of classifier is retrained on D_L[170].

How to identify useful valuable instances to query and learn from, and what sampling criteria are used?

In the following subsections we are going to present the steps of the approach, starting first of all with the sampling phase.

Using an AL selection strategy unlabeled instances deemed to be valuable are selected, selection strategy is based on sampling criteria. For our ASSM, we propose two selection strategies, we use two unsupervised clustering algorithms, namely K-means for hard clustering and Fuzzy C-Means(FCM) for soft clustering. The two selection strategies are thereafter compared to identify which clustering algorithm is able to select most valuable instance.

Step 1: *Clustering the unlabeled data set*

Clustering is an approach that allows to find patterns in the unlabeled data set in order to divide it into subsets(clusters), so that instances within the same cluster share the same characteristics. K-means and FCM are used for clustering. Both clustering algorithm have a common parameter that needs to be initialized at the very beginning, it is K the number of clusters. To choose the optimal number of cluster we use the elbow method[171].

A- ***K-mean*** the global K-means clustering algorithm can be outlined with the following equation:

Suppose we have a given data set $X=\{x_1, x_2, ..., x_N\}$, $x \in \{R^d\}$,

The M-clustering problem aims to partition the data set into M disjoint subsets(clusters) $C_1, ..., C_M$, while optimizing a clustering criterion. The most commonly clustering criterion employed, is the sum of squared Euclidean distances, between each data point x_i and the centroid m_k of the cluster C_k which contains x_i. This criterion is known as clustering error [172], and depending on the clusters centers $m_1, ..., m_M$.

$$E(m_1, ..., m_M) = \sum_{i=1}^{N} \sum_{k=1}^{M} I(x_i \in C_k) \parallel x_i - m_k \parallel^2, \tag{4.1}$$

Where I(x)=1 if X is true, and 0 otherwise.

A locally optimal solution is found with respect to the clustering error.K-means starts with centroids placed arbitrary, and proceed by moving at each step to minimize clustering error.

B- **Fuzzy C-means:** as a second AL based clustering, we choose a soft clustering algorithm Fuzzy C-means (FCM). The FCM algorithm attempts to partition an ensemble of elements $X=\{x_1, x_2, ..., x_N\}$ into an ensemble of fuzzy clusters. FCM clustering algorithm can be outlined with the following equation:

Given a finite ensemble of data , FCM returns a list of c centroids $C=\{c_1, c_2, ..., c_c\}$ and a partition matrix.

$W=w_{i,j} \in [0,1], i = 1, ..., n, j = 1, ..., c$,$w_{i,j}$ tells the degree of membership to which an element x_i belongs to cluster c_j.

The FCM algorithm aims to minimize an objective function[173]:

$$argmin_c \sum_{i=1}^{n} \sum_{j=1}^{c} w_{i,j}^{m} \parallel x_i - c_k \parallel^2, \tag{4.2}$$

Where:

$$w_{i,j} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \tag{4.3}$$

FCM is a fuzzy version of K-means algorithm, allowing each data point to have a degree of membership in all clusters rather than a distinct membership to just one cluster [174]. In other words, one data point can belong to two clusters or more,

Step 2: *Selection of valuable instances*

As explained earlier, only samples that would enhance the training set are selected(samples selection). Once the clustering done and the clusters are defined we start building the batch of instances that will be sent for query (Figure 4.2and Figure4.3).

4.2.1.2 *Sampling criteria*

We consider the three following criteria to select instances for both selection strategies K-means and FCM:

1. ***Informativeness :*** *for the K-means strategy*, the informative instances are the distant data points from the centroid of the cluster, also called sometimes outliers[175]. Outliers are considered meaningful, as they indicate rare events and can present critical actions, this makes these instances rich in information. We calculate the distance between each data point in the cluster and the centroid C_K to obtain a mean distance ***(Md)***, If the distance between one data point and the centroid is greater than to P times the mean: $Distance(x, C_k) > P * $ ***(Md)*** the data point in question is considered an outlier.

   For FCM strategy the informative instances are selected using the degree of membership.

   Given a finite set of unlabeled instances $X = x_1, ..., x_n$, FCM algorithm returns a list of C clusters $C = C_1..., C_c$ and a partition matrix.

   $$W = w_{i,j} \in [0, 1], \text{ i=1,...,n, j =1,...,c.}$$

   Each element $w_{i,j}$ is the degree to which an instance x_i belongs to a cluster c_j.

   For any instance x_i, and any cluster c_j if $w_{i,j} \in]0, 1[$ this means that x_i belongs to more than one cluster, and it is considered as an informative sample;

   2.Representativeness : for both selection strategies, the representative instances are the same. The goal is to select instances representing the majority; instead of sending all the data points of the cluster or randomly selecting instances to be labeled, each cluster will be represented by its centroid, this will also help to minimize the computational cost;

   3.Diversity : while selecting representative and informative samples, we are directly creating diversity assuming that samples from different clusters can be considered divers. This will allow to maximize the training utility of the selected batch sent to query[71].

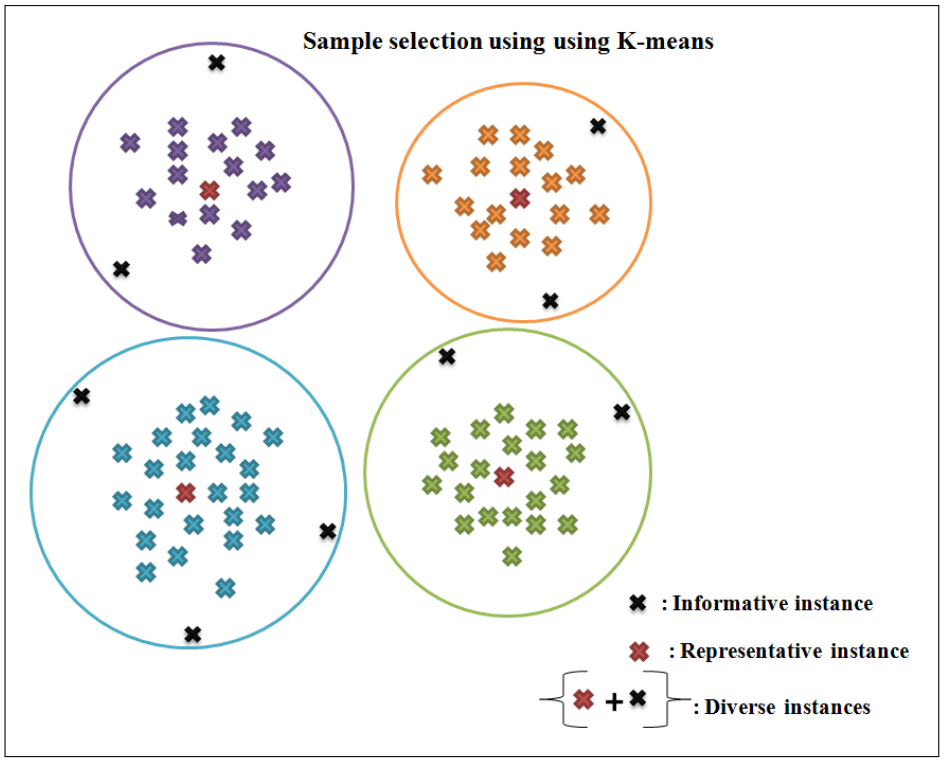

Figure 4.2 – *Sample selection using K-means.*

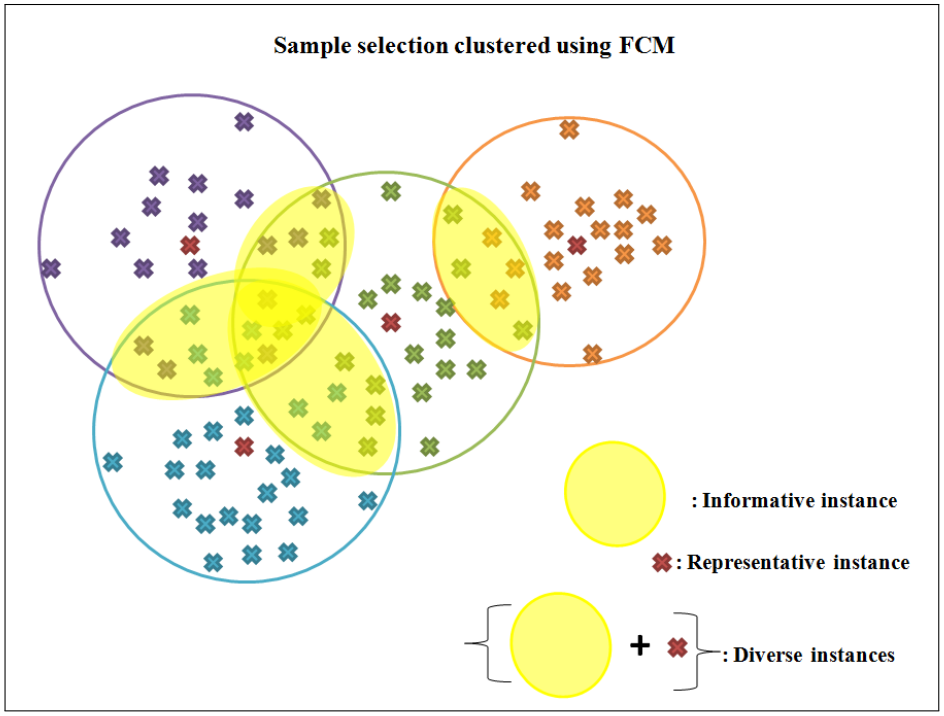

Figure 4.3 – *Sample selection using FCM.*

Once arrived at this stage, the batch of selected instances is ready to be sent for query, now it is the turn of the learning phase to take place.

4.2.2 Learning phase

In order to improve the reliability of the classifiers, especially when the labelled samples are limited, it is necessary to extend the initial training set. During this phase, the selected instances are labeled using an inductive semi-supervised learning model. A set of self-trained classifiers is used.

Step3 : *Query the labels of the selected instances*

The ensemble of classifiers labels the query batch using majority voting. We have chosen to use an ensemble of classifiers, also known as ensemble learning and Multi Classifier System (MCS), to cope with the following challenges:

1. Since we adopted semi-supervised learning as the general framework of our architecture, it is well known that the labeled data points that make up the training set are limited and that models trained on a small number of observations tend to over-fit and produce inaccurate results. One of the most commonly used methods to avoid over-fitting is the combination of several models, where a final prediction calculated as a weighted average of the predictions of various individual models will have a significant lower variance and can improve generalizability,
2. The goal of using multiple classifiers that adopt algorithms with varying strengths and weaknesses is to create a system that can achieve greater accuracy and outperform individual algorithms within the system itself[161],
3. The aim is to create diversity by using a Heterogeneous Ensemble method that will produce a lower error rate than using an individual classifier[176].

This diversity is created by using different learning algorithms to train each classifier, and this has been recognized as a very important feature in the combination of classifiers.

The individual decisions of each classifier are combined using hard Majority voting to arrive at a consensus decision[160]. At this stage we can retain the batch of the labeled instance, but we prefer to further refine the labeled instance, before storing them in the training set (the case base of the CBR system),

Step4 : *Selection of informative labeled instances*

Using majority voting, predictions made by the ensemble of classifiers are combined. We consider the highest numbers of votes, assigning to an instance the label that most classifiers agree on. Assuming that C_t presents classifier in an ensemble E, where $t = \{1, 2, 3, ..., n\}$. The decision of the t^{th} classifier C_t is donated by $d_{tj} \in \{0, 1\}$, where $j = \{1, 2, 3, ..., k\}$ and k is the number of classes [177]. The dicison produced will be $d_{t,j}$=1, if the t^{th} classifiers decides for class c_j, and $d_{t,j}$=0 else ways. The following equation outlines the output of the ensemble in majority voting:

$$\max_{1 \leq j \leq k} \sum_{t=1}^{n} d_{t,j} \tag{4.4}$$

In an ensemble learning, the set of classifiers is considered to be individual experts, and uncertainty can be determined as follows: if two individuals in the ensemble agree with a high level of confidence to label a sample, but the third classifier submits another label, then the sample is added to the set labeled with the label of the classifiers who agree. This leads to a combination of majority voting and uncertainty sampling[154]. The reasoning behind this selection strategy admits two aspects: first, samples for which uncertainty exists are considered informative since they modify the classifier as opposed to samples for which all classifiers agree. Second, the agreement between the two classifiers considered ”experts” makes it more likely that they have made the right decision,

Step5 : *Retain of selected labeled instance in the training set,*

Step6 : *Remove the selected instances from the unlabeled data set,*

Step7 : *Re-train ensemble of classifiers on updated training set.*

4.2.3 Stopping criterion

Most SSL methods are heuristics, modifications are introduced on standard supervised learning algorithms, where unlabeled data is considered along with a small labeled data set. Indeed, a stopping criterion or a threshold is needed to reflect the confidence that the unlabeled data could be labeled correctly and used along with the already labeled data to train accurate classifiers. For our algorithm we considered a stopping criterion, to monitor the performance of the training model after new cases are added, and stop automatically if degradation is rated. Semi-Supervised Learning is a reliable technique when training data are scarce, but there is no guarantee that all available unlabeled data are always useful, so monitoring the performance of the training model after new cases are added to the overall training is necessary.

Some stopping criterion stop the learning when the selection algorithm for instance does not have any good candidates to select[178]. A frequently used stopping criterion for self-training models, stops if no labeled instances are moved from the set of unlabeled data Du To the training set DL.

A more sophisticated stopping criterion also used in self-training models, automatically stops learning when a degradation in the performance of the model is rated[179]. We consider the latter stopping criterion for our approach , at the end of each iteration, when a batch of newly labeled cases is retained in the case database, a PCC% is calculated , it represents the mean percentage of correct

classification[75].

$$PCC = \frac{\text{Number of well classified instances}}{\text{Total number of classified instances}} 100 \tag{4.5}$$

If the PCC% decreases, the self-learning procedure does not continue learning and the batch is not retained, the model learned in the previous iteration is considered the final one. Thus, the algorithm stops when a decrease in the performance is rated or when the unlabeled data set D_U is empty, and there are no instances to be selected.

The pseudo code of the proposed approach is presented in Algorithm 3.

Algorithm 3 Pseudocode of the proposed approach.

Given Unlabeled data (U), Labeled data (L), Ensemble of Classifiers (C), InitialBatch (I), FinalBatch (F), clustering algorithm.

1: **repeat**
2: Train ensemble of classifiers(C) using (L)
3: Cluster U using a clustering algorithm
4: Add valuable samples to I
5: Query the ensemble of classifiers on I
6: Select N instances with uncertainty predictions
7: Add N to F
8: $L = L \cup F$
9: Remove F from U
10: **until** stopping criterion is met

4.3 Results and Discussion

In this section we present the performance evaluation of our ASSM approach.

The data set used for the experiments are described in Section. In Section we present the parameters used to measure the performance of the approach. Finally in section we discuss the results obtained from the different experiments.

4.3.1 Data sets

For the experimental evaluation of the proposed approach, twelve(12) data sets were used.Nine(09) medical data sets, including eight(08) from the UCI Machine Learning Repository [180]: Breast Tissue, Breast Cancer Coimbra[181], Heart Failure Clinical Records[182], Breast Cancer Wisconsin, Blood Transfusion Service Center[183], Mammographic Mass[184], EEG Eye State, Cardiotocography. Another medical data set were used: Breast Pathologies[185]. Three other data sets from the UCI Machine Learning Repository were added for the evaluation, namely: Iris, Divorce Predictors[186], Glass Identification. These data sets contributed to the evaluation of many studies, a detailed description is presented in Table5.1.

Table 4.1 – *Description of data sets.*

Dataset	**#instances**	**#attributes**	**missing values**	**#classes**
Breast Pathologies	101	26	No	5
Breast Tissue	106	10	Yes	6
Breast Cancer Coimbra	116	10	Yes	2
Heart Failure Clinical Records	300	13	Yes	2
Breast Cancer Wisconsin (Diagnostic)	569	19	No	2
Blood Transfusion Service Center	749	5	Yes	2
Mammographic Mass	962	6	Yes	2
EEG Eye State	1500	15	Yes	2
Cardiotocography	2126	22	Yes	3
Iris	150	5	No	3
Divorce Predictors	170	55	Yes	2
Glass Identification	214	10	No	7

4.3.2 Experimental parameters (CB quality criteria)

As discussed in Chapter2, a case base is qualified effective if it can answer as much queries as possible, efficiently and correctly[40]. The evaluation of the CB quality can be made according to numerous criteria proposed in literature. For the evaluation of our approach, the *performance* criterion is used to measure the quality of the final CB. The objective of the maintenance approach is to start with a small case base, and throughout the iteration expand it with selected cases that are deemed useful, to maintain its quality as much as possible. Performance is characterized by the accuracy and a the number of cases stored in the case base [95]. Therefore, the following criteria are considered:

1. **Size (S%)** the average storage percentage: we measure the rate of increase in the size of the case base. The main objective of our ASSM method is to enlarge the case base starting with a small straining set by selectively saving new cases. The final percentage refers to the new case base size after adding new cases to the initial training set

2. **PCC (PCC%)** represents the mean percentage of correct classification over five-fold cross-validation.

$$S = \frac{\text{Number of final cases}}{\text{Size of training case base}} 100 \tag{4.6}$$

$$PCC = \frac{\text{Number of well classified instances}}{\text{Total number of classified instances}} 100 \tag{4.7}$$

To replicate the scenario of scarcity of labeled data, during the evaluation phase, the databases are divided into training and test sets. These sets are then used in the cross-validation, the ASSM starts with an initial training set containing 20% of labeled data from each data set. A five-fold cross-validation was used for all experiments (each fold was used as an independent training set, while the other four folds were used as a test set). Cross-validation is used to estimate the skill of the proposed approach and to ensure reliable results. Each training fold was manipulated using Bagging algorithm to create diversity among classifiers, as it allows to train our set of classifiers with different subsets (samples) of the existing small training set. A small change in the training set can result in a remarkable change in the classifiers output[163].

4.3.3 Results analysis

In the following subsections, we will first demonstrate, compare and discuss the performance of the two proposed selection strategies, namely the Active Learning based clustering methods: K-means and FCM. Next,in Table 4.3 we compare ASSM to random selection and standard CBR. In order to position our work in relation to the work proposed in literature, Table summarizes characteristics of some of the maintenance methods proposed in the literature operating at different levels of CBR. Finally, we evaluate the performance of the proposed approach when disposing of different portions of labeled data.

As presented in section4.2, the goal of the proposed ASSM approach is to retain new cases in the training set (case base), selected from unlabeled data set for their value, in order to learn new hypotheses to improve the reasoning of the CBR system. This retention is controlled, at the end of each iteration, after the final batch of selected cases has been stored the classification accuracy of the case base is measure. This produces a more compact case base with very satisfactory accuracy results, even starting with a very small training set representing one-fifth of the data set. Thus, the challenge of sparse labeled data can be overcome and we can demonstrate that a quality case base can be built even when starting with few labeled data.

Table 4.2 shows, for each dataset, the accuracy of the classification obtained according to the number of retained cases (storage size), Figure 4.4 and 4.5 shows the case base retention by the ASSM for every datasets in the test, using the two proposed selection strategies K- means and FCM.

Table 4.2 – *Comparing the performance of ASSM with two sampling strategies (K means, FCM).*

	ASSM			
	K-means		**FCM**	
	Size(%)	**PCC(%)**	**Size(%)**	**PCC(%)**
Breast Pathologies	83.90%	**94.16%**	82.71%	94.09%
Breast Tissue	73,75%	72,61%	55,31%	**73,91%**
Breast Cancer Coimbra	79,09%	**90,90%**	73,86%	89,72%
Heart Failure Clinical Records	66,00%	**81,71%**	57,74%	81,12%
Breast Cancer Wisconsin (Diagnostic)	71,92%	94,42%	78,81%	**95,78%**
Blood Transfusion	66,51%	**86,55%**	51,92%	86,16%
Mammographic Mass	76,23%	**91,18%**	50,25%	88,19%
Cardiotocography	72,52%	90,42%	76,41%	**93,63%**
EEG Eye State	57,68%	**98,31%**	70,38%	98,22%
Iris	68,08%	91,24%	48,27%	**92,85%**
Divorce Predictors	80,85%	**98,54%**	36,00%	95,02%
Glass Identification	54,73%	75,90%	61,94%	**77,34%**

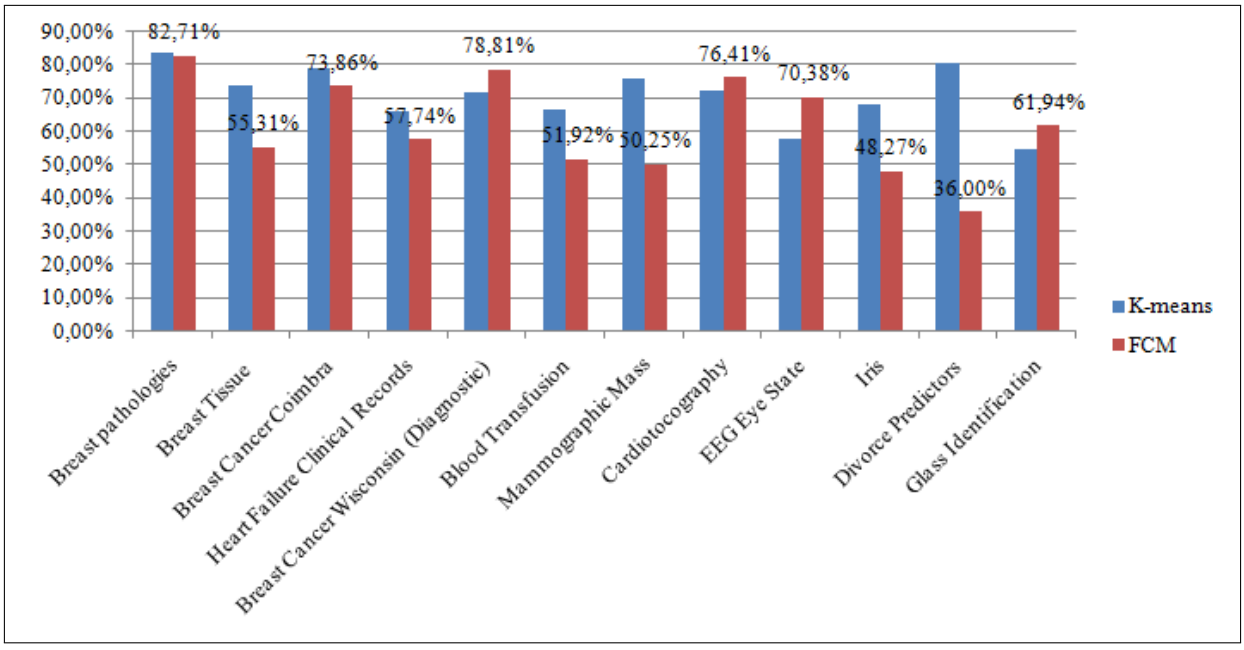

Figure 4.4 – *Case retention for ASSM(using K-means and FCM) for all datasets.*

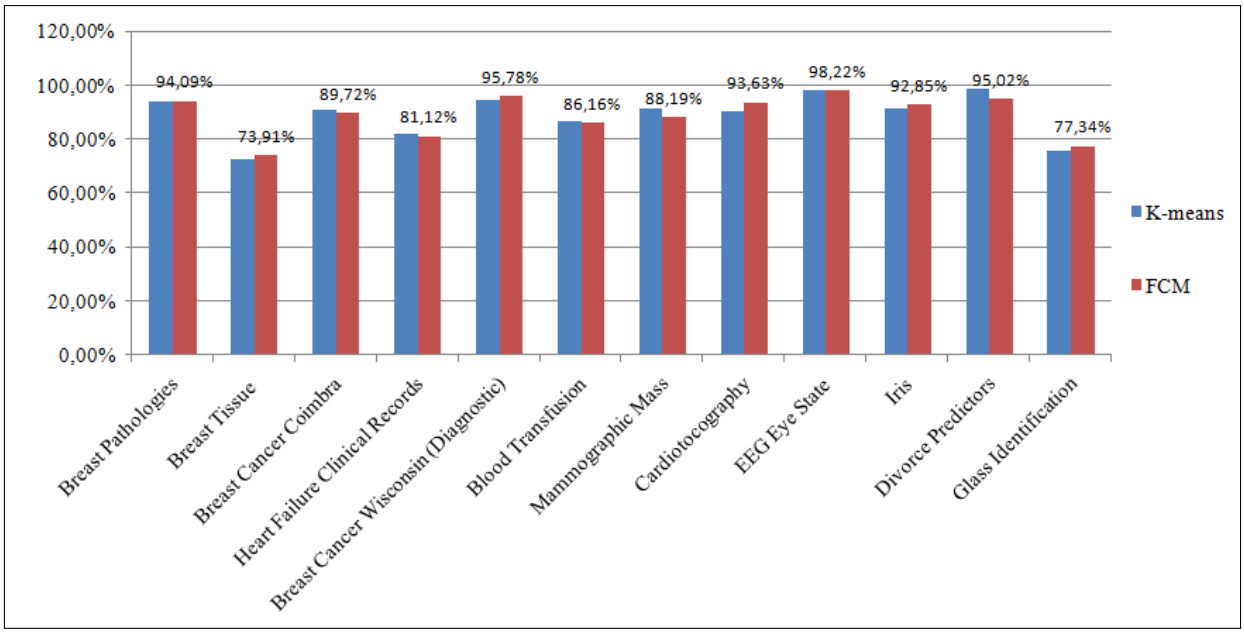

Figure 4.5 – *Classification accuracy for ASSM (using K-means and FCM) for all datasets.*

Two selection strategies were proposed for the ASSM sampling engine, and their ability to target valuable instances was compared. FCM and K-means achieved almost the same percentage of correct PCC classification (Figure 4.5). However, the storage size is much smaller with FCM than with K-means (Figure 4.4),this is due to the fact that both strategies are certainly based on clustering algorithm, but K-means is hard clustering and FCM is soft clustering,which means that the portioning of instances changes from one algorithm to another, as well as the selected instances.

For example, some data sets seem to be particularly well suited for the FCM method, which achieves better classification accuracy with a storage size 20% smaller than the K-means method, e.g. Breast Tissue and IRIS. The results prove that the proposed method is capable of producing a compact quality case base with good classification, starting with a remarkably small training set as a case base and expanding to a more competent case base. We attribute this success to ASSM property of targeting valuable instances, assigning labels to these cases, and retaining them in the case base to enhance learning.

For the rest of the experiments, we take into consideration the results obtained by one of our selection strategies, namely FCM, as it performed slightly better than K-means.

Since the cases stored in the CB are not only retained once labeled, but there is a whole selection process that takes place during the sampling phase and a second selection that takes place at the learning phase, after the assignment of labels to the instances. We can very well categorize our maintenance approach as a maintenance retention strategy. To demonstrate that our approach actually improves the quality of the CB, and that the classification rate increases not only because we increase the volume of the training set but mainly because we store useful and valuable cases that increase the value of the training set(the CB).

We compare the results obtained using our approach to random retention and a standard CBR retention policy. Random retention is a retention policy in which a random number of cases are selected from the unlabeled database and are directly retained in the case base once labeled. To do this, we randomly select and retain the exact number of cases retained by one of our selection strategy. The objective is to compare the contribution to the performance improvement of the CB, by a batch of randomly selected cases, and a batch of sampled cases encompassing the three sampling criteria. On the other hand, in the standard CBR retention, all cases are added to the case base as soon as they are assigned a label.

The results presented in Table4.3 show that indeed, not all cases can contribute to the improvement of case base learning in terms of accuracy and that it is possible to construct a smaller competent case base that achieves higher classification accuracy. Our retention strategy yields very good accuracy rates with smaller case bases, e.g. the Breast Tissue data set, ASSM generates a CB with a size of 55.31% with a classification accuracy of 73.91%, compared to 51.83% for random retention and 69.66% for standard CBR that retains all cases (100%). Similarly, for Heart Failure records, the final case base is of a size of 57.74% , with an accuracy of 81.12%, compared to 80.93% for standard CBR retention.

Table 4.3 – *Comparing performance of ASSM to random retention and standard CBR retention.*

	ASSM using FCM		**Random retention**		**Standard CBR retention**	
	Size(%)	**PCC(%)**	**Size(%)**	**PCC(%)**	**Size(%)**	**PCC(%)**
Breast Pathologies	82.71%	**94.09%**	82.71%	88.48%	100%	90.55%
Breast Tissue	55,31%	**73,91%**	55,31%	51,83%	100%	69,66%
Breast Cancer Coimbra	73,86%	**89,72%**	73,86%	87,08%	100%	85,34%
Heart Failure Clinical Records	57,74%	**81,12%**	57,74%	80,97%	100%	80,93%
Breast Cancer Wisconsin (Diagnostic)	78,81%	**95,78%**	78,81%	93,78%	100%	94,05%
Blood Transfusion	51,92%	86,16%	51,92%	85,74%	100%	**91,17%**
Mammographic Mass	50,25%	88,19%	50,25%	87,35	100%	**90,53%**
Cardiotocography	76,41%	**93,63%**	76,41%	91,04%	100%	91,91%
EEG Eye State	70,38%	**98,22%**	70,38%	95,12%	100%	96,03%
Iris	48,27%	**92,85%**	48,27%	71,82%	100%	90,08%
Divorce Predictors	36,00%	**95,02%**	36,00%	97,39%	100%	**97,64%**
Glass Identification	61,94%	**77,34%**	61,94%	72,40%	100%	74,69%

It is noteworthy that our approach represents a maintenance strategy that is executed during the development phase while building the case base, while maintenance strategies are usually applied during or at the end of the CBR life cycle. We were unable to identify an approach that is closest to ours, so we considered comparing our strategy with existing CBR strategies in the literature based on the performance criterion. In order to position our work in relation to the work proposed in literature, a selection of different CBM strategies is presented in Table4.4. The performance criterion is taken into consideration to evaluate the proposed maintenance strategies.

- The size(%) represents the final size of the generated CB (taking into account the fact that some strategies are incremental and others decremental),
- PCC (PCC%) represents the mean percentage of correct classification,
- The «Maintenance activity »refers to the the stage during which the maintenance is performed, knowing that the maintenance can be integrated «online »during the operation of the CBR system, «offline »outside the operation CBR life cycle, or at the development phase, which is the case for our proposed approach, where we maintain our CB while building it, to empower the reasoning process once the CBR system is operational.

The proposed maintenance approach can be evaluated by positioning it among the follwing well-known CBM strategies: CNN [86], ENN [87], RENN [88], Relative Performance (RP) [77], BBNR [91], GCNN [92], RDCL[94], Selective Retention (SR) [95], and RBM_{Cr} [74].
Amid these strategies, four share the same specification as ASSM, namely, CNN[86], Relative Performance(RP)[77], GCNN[92], Selective Retention(SR)[95]. These strategies are incremental algorithms, the learning model adapts to new data without forgetting its existing knowledge. A retention policy is envisaged in order to add cases to the CB according to certain criteria.

Table 4.4 – *Summary of different CBM strategies.*

Maintenance Strategies	**Performance criterion**		**Maintenance activity**
	Size(%)	**PCC(%)**	
Our approach	70,79%	88,83%	Development Stage
RBMCr[74]	78,60%	63,34%	Operating Stage
SR[95]	139,95%	81,69%	Operating Stage
RDCL[94]	88,72%	78,63%	Operating Stage
GCNN[92]	58,32%	75,69%	Operating Stage
BBNR[91]	82,70%	78,29%	Operating Stage
RP[77]	107,16%	79,63%	Operating Stage
RENN[88]	75,12%	77,55%	Operating Stage
ENN[87]	77,40%	77,25%	Operating Stage
CNN[86]	33,83%	73,78%	Operating Stage

Figure 4.6 and 4.7 show very promising generalized results for our approach. ASSM allows a better generalized classification accuracy 88.83% and satisfactory storage size rates of 70.94%, even when starting with a small case base as a training set due to the lack of supervised data unlike the rest of the strategies that work on a large case base from the outset. Although CNN and GCNN score a 33.83% and 58.32% of storage size rate but we need to emphasize on the significant reduction of the classification accuracy.

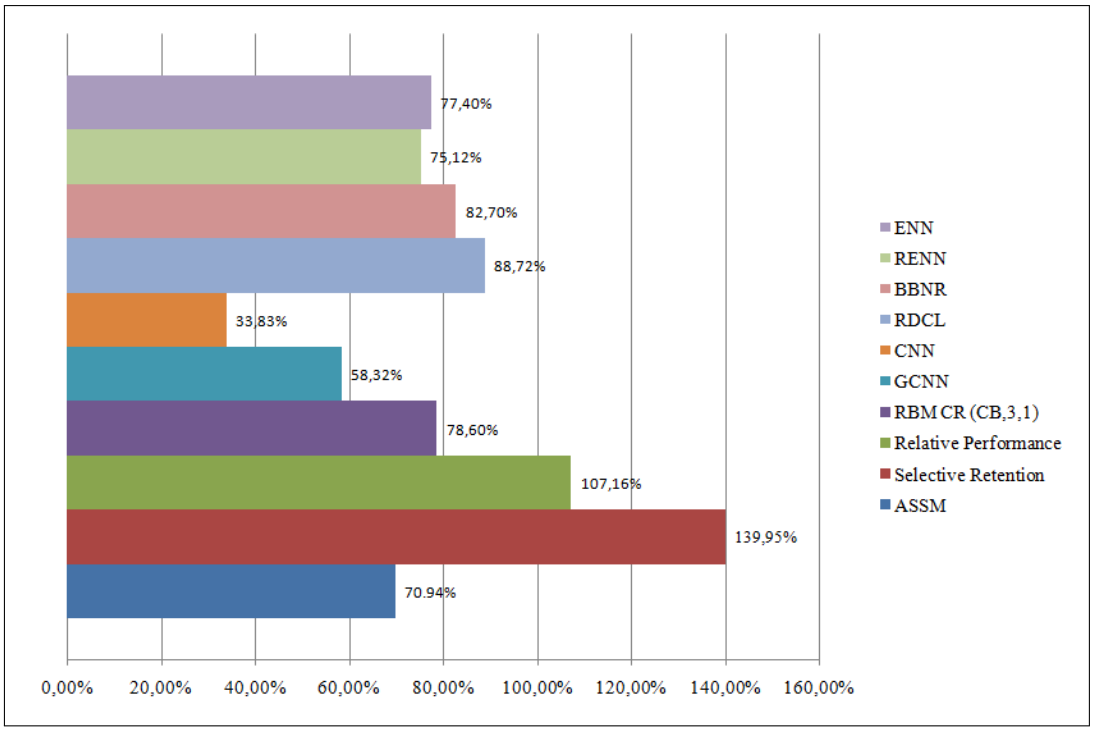

Figure 4.6 – *Comparison of ASSM storage size (%) to state-of-the-art strategies.*

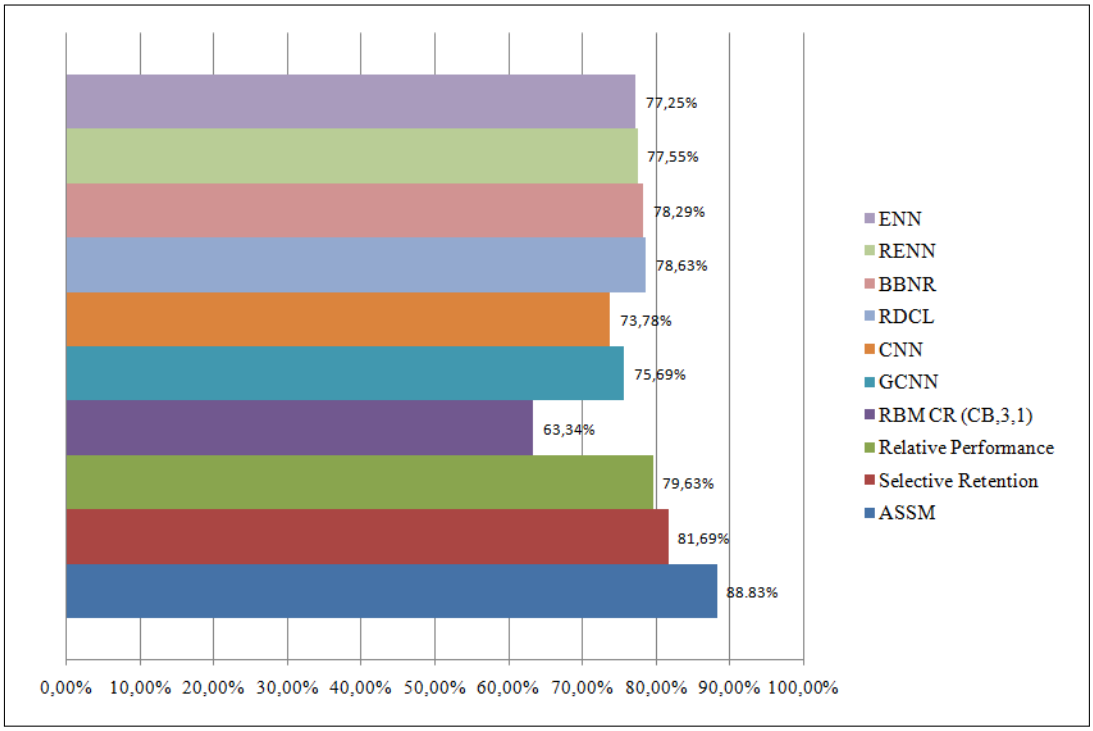

Figure 4.7 – *Comparison of classification accuracy (%) of ASSM to state-of-the-art strategies.*

The main objective of our ASSM approach is to maintain a small but useful number of cases that contribute to the informativeness, representativeness and diversity of the case base. Accordingly, we avoid indiscriminate growth in the size of the case base, which can lead to a degradation of the quality of the system.

All this while achieving satisfactory results in terms of classification accuracy. This result is even more significant when the storage size (S%) is taken into account: ASSM stores (on average) 70.94% of the cases(Figure4.6), and is still able to provide good classification accuracy rates(Figure4.7).

4.4 Conclusion

Our goal is to build and maintain a case base during the CBR development phase, to overcome the difficulty of assembling labeled case bases, traditionally assumed to exist or determined by human experts. We select valuable instances from a pool of unlabeled data using sampling criteria, which would improve the quality of the case base and thus the performance of the CBR system once operational.

In this study an Active Semi-Supervised Maintenance (ASSM) strategy is proposed, using machine learning techniques to cope with the problem of the scarcity of labeled cases. ASSM monitors the selection and retention of cases in the case base at two stages:

1. Sampling phase: using active learning,
2. Learning phase, using self-training, an inductive semi-supervised learning technique.

We start with an initial phase, namely the sampling phase, during which unsupervised learning methods were adopted in order to target valuable instances. For this phase we opted for a clustering based Active Learning, in order to study the underlying structure of unlabeled data set. Two sampling strategies were proposed and later on compared, namely a hard clustering algorithm: K means, and a softer clustering algorithm: FCM.

A cluster analysis of the generated clusters is performed, to discover the different type of instances within each cluster, and then select valuable instance using sampling criteria to be send for query. For our study we combined three sampling criteria: *informativeness*, *representativeness* and *diversity* (the last criteria is generated when combining both informative and representative samples in the same query batch), these sampling criteria are used to increase the chances of building a batch of useful instances, that will be an added value to the case base once labeled and stored.

Once the query batch is send to be labeled at the learning phase, an inductive semi-supervised learning method is used t assign labels to the selected instances, using a set of self trained ensemble of classifiers.

Performance criterion was considered to evaluate the quality of the case bases.

The evaluation of the proposal demonstrates the effectiveness of ASSM which is interesting as a CBM strategy, able to be efficient in terms of a controlled growth of the storage size and scoring satisfying classification.

The performance boost is due to the quality CB updated at each iteration with informative, representative and divers data points. This enhances the learning of classifiers trained each iteration with the updated training set (CB). Queries are unlabeled data points selected according to three sampling criteria as explained in section 4.2.1.2, then send to be labeled by the ensemble of classifiers. Rather than adding the entire labeled set, we again select the most informative samples following the majority voting. Informative labeled instances are retained to provide a richer set to train the classifiers for a better precision. This is a crucial step to obtain a good classification. It is recognized that the goal of using multiple classifiers that adopt algorithms with varying strengths and weaknesses is to create a system that can achieve greater accuracy and outperform individual algorithms.

Case Base Maintenance: Clustering Informative, Representative and Divers cases (C_IRD)

Contents

5.1 Introduction

As discussed in Chapter2 , case-based reasoning systems are implemented to operate over a long period of time, supporting the active learning of new cases through the retention phase of the classical CBR life cycle. A long-term CBR application ultimately leads to the recognition of the importance of maintaining the case-based reasoner.

The retention of cases in the knowledge container at the end of each CBR life cycle leads to a very fast growth of the latter, which can negatively affect the quality of CBR system, and results in a slow execution of queries in the retrieve phase. This performance degradation is due to memory swamping or exposure to harmful experiences[68], both factors affect the overall utility of a CBR system.

To avoid performance degradation, CBR systems need to be maintained, and various works have been proposed to address the mentioned challenges (section2.4.2). All of them have the same objective, namely, to ensure and improve an efficient CBR process. Case Base Maintenance has been outlined as the process of improving the performance of CBR systems: «Case base maintenance implements policies for revising the organization or content of the case base to facilitate future reasoning for a particular set of performance objectives »[44].

In the previous chapter4, we presented a CBM strategy dedicated to the development stage of the CBR system, where we simultaneously build and maintain a quality CB to be used later on as a reasoning core in CBR systems. However, after several runs of CBR life cycle and storing newly solved cases in the CB at the retain phase for future use, the quality of the CB can be affected by the uncontrolled growth of it, we have discussed this point in chapter2 and the impact it can have on the overall quality of the CB which directly involves the quality of the system.

Seeking to maintain or even improve the quality of the case base built during the development phase which might have degraded after several CBR reasoning cycles, we were interested in one particular branch of research. This branch focuses on the partitioning of the case base which builds an elaborate CB structure and maintains it continuously(see section2.4.2.1). In this chapter, we propose a maintenance approach that addresses the drawbacks that follow an operational CBR system and the blind retention of cases.

The proposed approach focuses on balancing the efficiency of case retrieval and the competence of a case base by employing a soft clustering technique **FCM**. The method could be able to maintain the case bases giving satisfactory accuracy, reducing its size, which leads to a reduction of retrieval time. Following the use of FCM in our first contribution, and the latter obtaining significantly better results than Kmeans a hard clustering algorithm, when both used to target valuable instances. We were interested in further exploring the potential of FCM as a maintenance strategy.

5.2 Proposed approach: Clustering Informative, Representative and Divers cases (C_IRD)

The main objective of CBM is to reduce the size of the case base while maintaining or even improving the quality of the CBR system as much as possible. Maintaining or improving the quality of the system depends on the quality of its case base, a knowledge container that needs to be given full attention. When we focus on the quality of the system, we are required to carefully select cases for deletion, without depreciating the efficiency of CBR and the competence of the case base[75].

Building a high-quality case base requires a CBM strategy that: delivers a small CB size, removes irrelevant cases from the case base, and targets only valuable cases to be retained to increase classification accuracy. We use a clustering algorithm for our proposed approach to cluster the initial CB into smaller clusters, to enable the identification of valuable cases using sampling criteria, traditionally used in Active Learning. We determined three types of valuable cases .

5.2.1 Soft Clustering to target valuable cases to retain:

In an attempt to target only the valuable cases to be maintained in the case base, we use a soft clustering algorithm to divide the original case base into smaller clusters, in order to facilitate the identification of cases deemed valuable.

FCM algorithm attempts to partition an ensemble of elements $X=\{x_1, x_2, ..., x_N\}$ into an ensemble of fuzzy clusters. FCM clustering algorithm can be outlined with the following equation:
Given a finite ensemble of data , FCM returns a list of c centroids $C=\{c_1, c_2, ..., c_c\}$ and a partition matrix.
$W=w_{i,j} \in [0,1], i = 1, ..., n, j = 1, ..., c$,$w_{i,j}$ tells the degree of membership to which an element x_i belongs to cluster c_j.
The FCM algorithm aims to minimize an objective function[173]:

$$argmin_c \sum_{i=1}^{n} \sum_{j=1}^{c} w_{i,j}^{m} \parallel x_i - c_k \parallel^2, \tag{5.1}$$

Where:

$$w_{i,j} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \tag{5.2}$$

5.2.2 Which cases should be retained and why?

In order to have a good CB quality, after clustering the case base using FCM algorithm, we should retain valuable cases whose deletions directly reduce the competence of the system, namely (See Figure5.1):

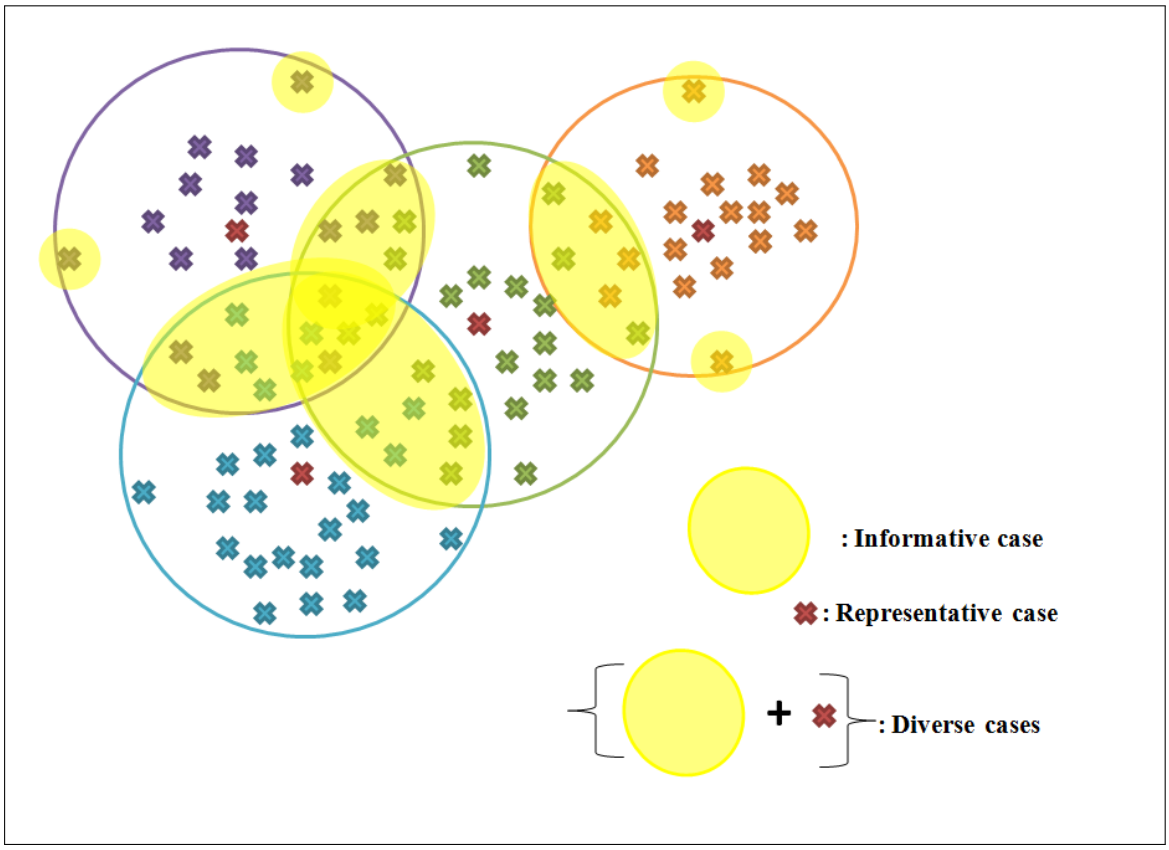

Figure 5.1 – *C_IRD valuable cases to retain in the CB.*

1. *Informative cases:* these cases add rich information to the CB, and we target them using two methods:

   (1) an informative case is a case with no other cases within the CB similar to it. This type is referred to as outliers and also known as isolated cases, which should be retained, as their removal directly diminishes the competence of the system,

   (2) in informative cases is also selected using a degree of membership, two versions of FCM are used for the selection of informative cases using the memebership value:

- The first version is referred to as C_IRD_1 where a case that belongs to more than one cluster is considered as an informative case,
- The second version is referred to as C_IRD_2, using the membership value, we try to remove redundant cases, we further refine the selected cases by checking if there areany redundant cases with the same membership value, and then select only one of these cases to retain.

2. *Representative cases:* a representative case is a case among a group of similar cases, the case with a high representation is selected, which means that it has a high density, so it represents more neighbouring,

3. *Diverse cases:* diversity aims at keeping cases that scatter the entire input space, instead of focusing on one small region of it. Diversity cases are a direct consequence of the retention of informative and representative cases. By keeping representative and informative cases in the case base, we directly create diversity by assuming that cases from different clusters can be considered diverse.

These sampling criteria are discussed in detail in the previous Chapter4 at section4.2.1.2.

After we partition the original case base using FCM an unsupervised soft clustering algorithm, and select the valuable cases (informative, representative and diverse), for each cluster, the cases that were not selected are removed. The removed cases simply consume space within the case base and increase the response time of the CBR system. Accordingly, as an attempt to maintain or improve the quality of the original case base, by applying our C_IRD method we build a new reduced case base that includes the cases that do not decrease the competence of the system.

5.3 Results and Discussion

For the experimental evaluation of our C_IRD approach, it is tested on five case bases from the U.C.I. machine learning repository data sets [180], considering only numerical data. These data sets have contributed to the evaluation of many studies, a detailed description of which is presented in Table5.1.

Table 5.1 – *Case Bases Description.*

Case Base	**#instances**	**#attributes**
Iris	150	4
Breast Cancer Wisconsin (Diagnostic)	569	19
Blood Transfusion Service Center	749	5
Mammographic Mass	962	6
Yeast	1483	8

Regarding the evaluation parameters, specifically the first two evaluation criteria, namely the storage size (S%) and the percentage of correct classification (PCC%), these are the same criteria used for the evaluation of the first approach (Chapter4,section4.3.2).

The goal of our C_IRD approach is to reduce the case base size while maintaining as much as possible the competence of the system. Thus, the following main criteria will be considered:

1. **Storage size (S%):** average storage percentage: we measure the rate of decrease in the size of the case base. The main objective of C_IRD method is to reduce the case base. The final percentage refers to the percentage of cases that were included in the original CB and are in the new reduced case base.
2. **Percentage of correct classification (PCC%):** the main applications in CBR use for the retrieve phase the nearest neighbor algorithm [187] technique.It is

a simple approach whose goal is to calculate the similarity between the cases stored in the CB and the new input case.

Therefore, we choose to select the 1-Nearest-Neighbor (1-NN) method to calculate the percentage of correct classification: The training set contains 80% of the case base and the test set contains the remaining 20% of the cases. For each data set We apply the 1-NN algorithm with a five-fold cross-validation task to compute the percentage of correct classification,

3. **Time:** the retrieval time in seconds : the goal of C_IRD method is to reduce the case base and thus the search time. Therefore, we choose the criterion "Time" to highlight the performance of our method in the reduction of retrieval time[75].

As we have already mentioned, the objective of our C_IRD approach is to maintain the case base by preserving or even improving their performance (Retrieval time) and competence (Storage size and Accuracy) during problem-solving. Thus, for the first experiment, we propose to evaluate the effectiveness of C_IRD compared to the results obtained with the initial non-maintained case bases, which we will refer to as (CBR) (Table5.2).

Table 5.2 – *Comparing CBR to the two versions of C_IRD.*

Datasets	**CBR**			**C_IRD1**			**C_IRD2**		
	S(%)	**PCC(%)**	**Time(s)**	**S(%)**	**PCC(%)**	**Time(s)**	**S(%)**	**PCC(%)**	**Time(s)**
IRIS	100%	93,06	0,454	55,4	91,35	0,052	12,13	64,07	0,034
Breast-W	100%	93,38	0,623	67,51	94,58	0,238	67,25	94,05	0,236
Blood-T	100%	71,60	0,354	4,34	79,90	0,11	2,97	78,84	0,090
Mammographic	100%	75,94	0,593	3,96	83,41	0,204	3,62	53,17	0,081
Yeast	100%	51,37	0,515	8,51	47,42	0,212	6,42	34,07	0,105

It is worth discussing the results obtained with the two versions of FCM used

at this point, we can observe that in terms of size reduction, C_IRD2 reduces the size of the case base more than C_IRD1. This is due to the fact that the latter removes cases that have the same degrees of membership, in the view that these are redundant cases, but this simply reduces the classification rates. It is necessary

to emphasize that the cases belonging to several classifiers should not be concerned by the deletion, these are cases rich in information (informative) and even if the degrees of membership are the same for some cases, it does not necessarily mean that we are dealing with a redundancy. The size reduction difference between C_IRD1 and C_IRD1 varies between 0.26% and 43.12%. This reduction in size results in a reduction of accuracy rates (between 0.13% to 30%), which supports our hypothesis that the cases belonging to several clusters are very significant for learning and should not be deleted.

We are conscious that our approach has shortcomings, but there are positive points that can be explored to improve C_IRD1. Therefore, we have decided to compare C_IRD to other well-known reduction techniques in the literature (Table 5.3, Table5.4 and Table5.5): CNN [86], RENN [88], ENN [87] and Instance Based learning IBL schemes[188] on the data sets presented above.

Table 5.3 – *Comparing storage size S (%).*

Datasets	**C_IRD1**	**CNN**	**RNN**	**ENN**	**IB2**	**IB3**
IRIS	55,4	27,63	93,33	95,33	24,00	24,00
Breast-W	67,51	16,3	16,87	81,69	35,48	30,46
Blood-T	4,34	37,3	38,72	32,1	26,09	26,00
Mammographic	3,96	64,21	54,26	82,52	53,48	53,93
Yeast	8,51	12,34	14,02	69,59	69,79	69,98

Table5.3 shows that our proposed method did not yield the best reduction rate on all case bases, namely IRIS and Breast-W where it retains respectively 44,6% and 32,49% of the initial case base, compared to other methods, for example: IB2 and IB3 which retain only 24,00% of IRIS CNN which retain only 16.3% for Breast-W. This is due to the fact that these two bases are too small, comparing to other case bases.

For the other case bases obtained, the sizes are generally reduced by more than 95%, compared to the initial sizes of the CBRs that contain all the instances (100% of the cases). For example, C_IRD1 removes more than 96.04% of the cases for the "Mammographic" case base.

Table 5.4 – *Comparing classification accuracy PCC (%).*

Datasets	**C_IRD1**	**CNN**	**RNN**	**ENN**	**IB2**	**IB3**
IRIS	91,35	73,00	94,23	91,60	91,67	91,67
Breast-W	94,58	68,18	67,05	94,66	69,69	70,56
Blood-T	79,90	67,94	66,65	71,63	74,69	74,21
Mammographic	83,41	70,82	78,60	77,04	66,28	66,40
Yeast	47,42	83,56	83,92	88,08	73,82	73,30

The same remarks are valid for the accuracy, where the accuracy provided by our C _IRD1 shows slightly better values (see Table5.4). It is indeed better than that of CBR which retains all instances(Tbale5.1), especially for the "Mammography" case base. It reaches 83.41 % PCC, which is a significant difference compared to the one given by CBR: 75.94% PCC.

This proves that our method not only maintains the quality of the case base but also can improve it. Moreover, we observe that the PCC obtained by our C_IRD method is better than those obtained by the other well-known reduction techniques. For example, for the "Breast-W" case base, the PCC provided by our approach is higher than 94%, while it is lower than 71% for the other methods.

Table 5.5 – *Comparing retrieval time in seconds.*

Datasets	**C_IRD1**	**CNN**	**RNN**	**ENN**	**IB2**	**IB3**
IRIS	0,052	0,011	0,010	0,013	0,002	0,002
Breast-W	0,238	0,430	0,350	0,734	0,244	0,227
Blood-T	0,11	0,098	0,183	0,194	0,203	0,197
Mammographic	0,204	0,208	0,199	0,815	0,339	0,032
Yeast	0,212	0,640	0,604	0,134	0,595	0,581

Regarding the response time, as shown in Table 4, the results presented by C_IRD1 are better than those given by CBR. Our maintenance method reduces the retrieval time since the case bases have been reduced. For example, for the "Mammography" case base, since C_IRD1 retains only 3.96% of the cases, the retrieval time is about twice as better as CBR. Similar observations are made regarding the comparison with CNN, RNN, ENN and IBL in which C_IRD1 has, in general, the shortest retrieval time.

5.4 Conclusion

This chapter presents an approach that seeks to maintain or even improve the quality of the case base built during the development phase which may have degraded after several rounds of CBR reasoning. We have presented a case base maintenance approach called C-IRD, our approach uses a soft clustering algorithm to target the valuable cases to be saved in the case base while removing other less valuable cases, leading to better results in terms of classification accuracy hence CBR quality. The experiments show that our method is a promising case base maintenance approach able to be efficient in terms of reducing the size of the case base and the retrieval time and to obtain a satisfactory classification accuracy.

We recognize that our approach has shortcomings, but there are certain advantages that can be investigated in future work to improve the method, by further exploring membership values of cases belonging to many clusters. Furthermore, to show the scalability of our C_IRD approach, we intend to evaluate it on real case bases.

General conclusion and perspectives

The early notion of Case-Based Reasoning (CBR) emerged from the results of various research conducted on the human brain, and was heavily influenced by cognitive sciences. CBR is a branch of artificial intelligence that combines machine learning and reasoning techniques to solve new problems by adapting solutions that have worked for similar past experiences. Cases are generated from these prior events to build a knowledge container known as the case base. CBR, unlike other artificial intelligence problem-solving approaches, replicates human thinking and is memory-based, making it highly comparable to human reasoning. The quality of the case base, a critical knowledge container, is directly linked to the implementation of an effective CBR system. CBR is a memory-oriented cognitive model that focuses on how to acquire new skills or generate hypotheses for new situations based on previous experiences. This artificial intelligence approach relies heavily on the performance of the case base to make highly appropriate decisions. When employing a CBR framework to create a computer-aided diagnostics system, the quality of the case base is highly relevant. Therefore, case base maintenance becomes critical when considering the implementation a computer-aided diagnostic system using the case-based reasoning approach. The main objective of this work is to study an approach to build and maintain a case base during the early stages of the implementation of a CBR system, namely the development phase. We also propose an approach to maintain the quality of the case base acquired during the development phase once the CBR system is operational. Considering the potential degradation of the quality of the case base after several reasoning cycles that lead to a rapid growth of the case base. We present two contributions that have been

supported in this work, both represent case base maintenance strategies, each with different objectives, operating at a different level of the CBR system.

We began by presenting an approach that can be perceived as a preventive action, which qualifies it as a preventive maintenance strategy. In engineering preventive maintenance is a preventive action which aims to reduce the probability of failure of an element(in our study the case base) or the degradation of the performance of a service provided (in our scenario the service is reasoning), in order to prevent an equipment from failure (the CBR system can be perceived as an equipment) [22]. We are mindful of the problems that can arise using a limited case base, instead of launching a CBR system with a small case base, which may very likely lead to many reasoning mistakes and a poorly performing system, we address these potential problems before they occur, and prevent them from arising once the system is operational.

The first contribution consists of a maintenance strategy where machine learning techniques are used to build and enrich the knowledge container with relevant and useful cases for reasoning. The aim is to take advantage of unlabelled data, using active learning in conjunction with semi-supervised learning to select the instances judged as valuable from the pool of unlabeled data points. A pool of unlabeled data along with the few labeled data available are used in order to build a quality case base given the scarcity of such case bases supposed to exist or predefined by human experts, which is rarely the case.

The proposed ASSM maintenance approach was evaluated by positioning it among well-known CBM strategies and also standard CBR and random addition. Our approach achieves better generalized classification accuracy of 88.83% and satisfactory storage size rates of7 0.94%, even when starting with a small case base as a training set due to the lack of supervised data.

Since case-based reasoning system is built to run for long periods of time, adding cases to the case base through the retention phase. This results in a rapid expansion of the case base, which can negatively affect the quality of the CBR outcomes and can slow the speed of the query execution time at the retrieval phase.

To cope with this issue, a second maintenance strategy is proposed, in order to maintain or improve the quality of the case base built during the development phase, once the CBR system is operational. Given the risk of degradation of the quality of the case base after several reasoning cycles that lead to a rapid growth of the case base. Maintaining or improving the quality of the system depends on the quality of its case base, a knowledge container that needs to be given full attention. A high-quality case base requires a case base maintenance strategy that: delivers a small CB size, removes irrelevant cases from the case base, and targets only valuable cases to be retained to increase classification accuracy. We use a clustering algorithm for our proposed approach to cluster the initial CB into smaller clusters, to enable the identification of valuable cases using sampling criteria, traditionally used in Active Learning (Informativeness, rapresentativeness and diversity). The objective is to reduce the size of the case base using a soft clustering technique namely, FCM to identify the relevant cases that should be saved and those that should be removed from the case base.

The experiments show that our method is a promising case base maintenance strategy, capable of reducing the size of the case base and retrieval time while maintaining a high level of classification accuracy. We recognize that our approach has shortcomings, but there are certain advantages that can be investigated in future work to improve the method by looking into the membership values of cases belonging to many clusters.

Our proposed method did not yield the best reduction rate on all case bases, namely IRIS and Breast-W , this is due to the fact that these two bases are too small, comparing to other case bases. For the other case bases obtained, the sizes are generally reduced by more than 95% , compared to the initial sizes of the CBRs that contain all the instances (100% of the cases). For example, C_IRD1 removes more than 96.04% of the cases for the "Mammographic" case base. This proves that our method not only maintains the quality of the case base but also can improve it. Moreover, we observe that the PCC obtained by our C_IRD1 method is better than those obtained by the other well-known reduction techniques. For example, for the "Breast-W" case base, the PCC provided by our approach is higher than 94%, while it is lower than 71% for the other methods.

Regarding the retrieval time, the method reduces the latter since the case bases have been shortened. The retrieval time is about two times better than CBR, C_IRD1 has in general the shortest retrieval time.

Perspectives

The results obtained by our approaches are very pleasing and encourage us to pursue this research path. However, other directions remain to be explored including:

- Regarding the first contribution, we intend to introduce some changes to the approach in the interest of improving it as well as investigate the influence this may have on the results achieved. First, we would like to explore other techniques for combining classifiers and creating a multi classifier system, which is a rapidly emerging study area . The goal is to fulfill a crucial criteria for making an ensemble of classifiers more accurate than the individual classifiers that make it up, which is: diversity, as well as attempting to accomplish this by combining different classifiers than those used in our approach,

- As far as the key element of an active learning algorithm is concerned, which is the selective sampling strategy, for the sampling phase of our approach we opted for a clustering based active learning, in order to study the underlying structure of an unlabeled data set. Two sampling strategies have been proposed and subsequently compared, namely a hard clustering algorithm: K means, and a softer clustering algorithm: FCM. We would like to investigate the impact of missing values in unlabeled data and by further exploring the underlying cluster structure using FCM as a sampling strategy, as it has shown interesting results in terms of classification precision with small storage size. In addition, for future work we intent to explore alternative active learning scenarios and different sampling strategy options in addition to the clustering algorithms that are considered as a pool-based active learning technique,

- For the experimental evaluation of the proposed approach, twelve (12) datasets were used. Nine (09) medical datasets, including eight (08) from the UCI machine learning repository, which are datasets with numerical attributes. For future work, we are interested in testing our approach directly on image data bases,

- Regarding the second contribution, we plan to further develop and test the proposed deletion-based maintenance technique on a larger number of case bases. We also aim to address the membership values of cases belonging to many clusters, as we believe that these values can reveal more useful information about these cases, which can be benefited from to maintain the case base.

List of Publications

Publications in international journals

1. Chebli, A. et al. (2021). Case-Base Maintenance: an approach based on Active Semi-Supervised Learning. International Journal of Pattern Recognition and Artificial Intelligence, p. S0218001421510113. doi: 10.1142/S0218001421510113.

2. Chebli, A., Djebbar, A., Merouani, H. F. D. (2020). Improving the performance of computer-aided diagnosis systems using semi-supervised learning: a survey and analysis. International Journal of Intelligent Information and Database Systems, 13(2-4), 454-478.

Publications in international conferences

1. Chebli, A., Djebbar, A., Merouani, H. F. (2021, November).Case Base Maintenance: Clustering Informative, Representative and Divers cases (C IRD). 15th International Conference On Information Technology And Applications,DUBAI, UAE. (Accepted for oral presentation).

2. Chebli, A., Djebbar, A., Merouani, H. F. (2018, November). Semi-Supervised Learning for Medical Application: A Survey. In 2018 International Conference on Applied Smart Systems (ICASS) (pp. 1-9). IEEE,Medea,Algeria.

3. Chebli, A., Djebbar, A., Merouani, H. F. (2018). A review: Case Base Maintenance based on case addition policies. 6th International Conference on Multimedia, computing Systems (ICMCS), Rabat, Morocco.

Publications in national conferences

1. Chebli, A., Djebbar, A. and Merouani, H. F. (2018) 'Semi-Supervised Framework for the Learning of CBR:Medical Data Application', in. 1st Conference on Informatics and Applied Mathematics IAM'2018, Université 8Mai 1945,Guelma,Algérie.

Bibliography

[1] Mohamed Karim Haouchine. *Remémoration guidée par l'adaptation et maintenance des systèmes de diagnostic industriel par l'approche du raisonnement à partir de cas.* PhD thesis, Université de Franche-Comté, 2009.

[2] Michael M Richter. The knowledge contained in similarity measures,"invited talk: the first international conference on case-based reasoning", sesimbra, portugal. 1995.

[3] Beatriz López. Case-based reasoning: a concise introduction. *Synthesis lectures on artificial intelligence and machine learning*, 7(1):1–103, 2013.

[4] Michael M Richter and Rosina O Weber. *Case-based reasoning*. Springer, 2016.

[5] Agnar Aamodt and Enric Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, 7(1):39–59, 1994.

[6] A Mille. Tutorial cbr: Etat de l'art de raisonnement à partir de cas. *Plateforme AFIA*, 99, 1999.

[7] Thomas Roth-Berghofer and Ioannis Iglezakis. Six steps in case–based reasoning: Towards a maintenance methodology for case–based reasoning systems. In *Proceedings of the 9th German Workshop on Case-Based Reasoning*, pages 198–208. Citeseer, 2001.

[8] Mehmet Göker and Thomas Roth-Berghofer. Development and utilization of a case-based help-desk support system in a corporate environment. In *International Conference on Case-Based Reasoning*, pages 132–146. Springer, 1999.

[9] Klaus-Dieter Althoff. Case-based reasoning. In *Handbook of Software Engineering and Knowledge Engineering: Volume I: Fundamentals*, pages 549–587. World Scientific, 2001.

[10] Eduardo Lupiani, Jose M Juarez, Jose Palma, and Roque Marin. Monitoring elderly people at home with temporal case-based reasoning. *Knowledge-Based Systems*, 134:116–134, 2017.

[11] Barry Smyth and Mark T Keane. Adaptation-guided retrieval: questioning the similarity assumption in reasoning. *Artificial intelligence*, 102(2):249–293, 1998.

[12] Mohamed-Karim Haouchine, Brigitte Chebel-Morello, and Noureddine Zerhouni. Competence-preserving case-deletion strategy for case-base maintenance. In *ECCBR'08*, volume 1, pages 171–184, 2008.

[13] Jose M Juarez, Susan Craw, J Ricardo Lopez-Delgado, and Manuel Campos. Maintenance of case bases: current algorithms after fifty years. IJCAI, 2018.

[14] Loukas Serafeim. What is machine learning: Supervised, unsupervised, semi-supervised and reinforcement learning methods. https://towardsdatascience.com/what-is-machine-learning-a-short-note-on-supervised-unsupervised-semi-supervised-and-aed1573ae9bb, 2020. Accessed: 2021-07-26.

[15] C Yones, Georgina Stegmayer, and Diego H Milone. Genome-wide pre-mirna discovery from few labeled examples. *Bioinformatics*, 34(4):541–549, 2018.

[16] Mohamed Farouk Abdel Hady and Mohamed Farouk. *Semi-supervised learning with committees: exploiting unlabeled data using ensemble learning algorithms.* Südwestdeutscher Verlag für Hochschulschriften, 2011.

[17] Eugene Charniak and Robert Goldman. Plan recognition in stories and in life. In *Machine Intelligence and Pattern Recognition*, volume 10, pages 343–351. Elsevier, 1990.

[18] Asma Chebli, Akila Djebbar, and Hayet Farida Djellali Merouani. Improving the performance of computer-aided diagnosis systems using semi-supervised

learning: a survey and analysis. *International Journal of Intelligent Information and Database Systems*, 13(2-4):454–478, 2020.

[19] Shahina Begum, Mobyen Uddin Ahmed, Peter Funk, Ning Xiong, and Mia Folke. Case-based reasoning systems in the health sciences: a survey of recent trends and developments. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(4):421–434, 2010.

[20] Nabanita Choudhury and Shahin Ara Begum. A survey on case-based reasoning in medicine. *International Journal of Advanced Computer Science and Applications*, 7(8):136–144, 2016.

[21] John W Creswell and J David Creswell. *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage publications, 2017.

[22] Mostafa Anouar Ghorab and Farid Mokhati. Vers une approche de maintenance préventive des systèmes multi-agents. 2019.

[23] Ramon Lopez De Mantaras, David McSherry, Derek Bridge, David Leake, Barry Smyth, Susan Craw, Boi Faltings, Mary Lou Maher, MICHAEL T COX, Kenneth Forbus, et al. Retrieval, reuse, revision and retention in case-based reasoning. *The Knowledge Engineering Review*, 20(3):215–240, 2005.

[24] Stefania Montani, Lakhmi C Jain, et al. *Successful case-based reasoning applications*, volume 62. Springer, 2010.

[25] David B Leake. Case-based reasoning: experiences, lessons, and future directions. 1996.

[26] Therani Madhusudan, J Leon Zhao, and Byron Marshall. A case-based reasoning framework for workflow model management. *Data & Knowledge Engineering*, 50(1):87–115, 2004.

[27] Agnar Aamodt. Knowledge-intensive case-based reasoning in creek. In *European Conference on Case-Based Reasoning*, pages 1–15. Springer, 2004.

[28] Ralph Bergmann, Klaus-Dieter Althoff, Sean Breen, Mehmet Göker, Michel Manago, Ralph Traphöner, and Stefan Wess. *Developing industrial case-based*

reasoning applications: The INRECA methodology. Springer Science & Business Media, 2003.

[29] Roger C Schank. *Dynamic memory: A theory of reminding and learning in computers and people*. cambridge university press, 1983.

[30] Janet L Kolodner. *Case-based Reasoning: Proceedings of a Workshop on Case-Based Reasoning: Holiday Inn, Clearwater Beach, Florida, May 10-13, 1988*. Morgan Kaufmann, 1988.

[31] Klaus-Dieter Althoff, Stefan Wess, Joerg H Siekmann, and JG Carbonell. *Topics in Case-Based Reasoning*. Springer-Verlag, 1994.

[32] Manuela Veloso, Jaime Carbonell, Alicia Perez, Daniel Borrajo, Eugene Fink, and Jim Blythe. Integrating planning and learning: The prodigy architecture. *Journal of Experimental & Theoretical Artificial Intelligence*, 7(1):81–120, 1995.

[33] John McCarthy. What is artificial intelligence? 1998.

[34] Earl Hunt. Cognitive science: Definition, status, and questions. *Annual Review of psychology*, 40(1):603–629, 1989.

[35] John K Debenham. *Knowledge systems design*. Prentice-Hall, Inc., 1989.

[36] Ramon Lopez De Mantaras and Enric Plaza. Case-based reasoning: an overview. *AI communications*, 10(1):21–29, 1997.

[37] Ralph Bergmann, Janet Kolodner, and Enric Plaza. Representation in case-based reasoning. *Knowledge Engineering Review*, 20(3):209–214, 2005.

[38] Ian Watson and Farhi Marir. Case-based reasoning: A review. *Knowledge Engineering Review*, 9(4):327–354, 1994.

[39] Janet L Kolodner. Making the implicit explicit: Clarifying the principles of case-based reasoning. *Case-based reasoning: Experiences, lessons & future directions*, pages 349–370, 1996.

[40] Brigitte Chebel-Morello, Mohamed-Karim Haouchine, and Noureddine Zerhouni. Auto-incrémentation d'une base dysfonctionnelle de cas pour un système d'aide au diagnostic et à la réparation. In *3ème Edition du Colloque International Francophone sur la Performance et les Nouvelles Technologies en Maintenance, PENTOM'2007.*, pages sur–CD. FUCaM-Polytech (Mons), 2007.

[41] Thomas Reinartz, Ioannis Iglezakis, and Thomas Roth-Berghofer. Review and restore for case-base maintenance. *Computational Intelligence*, 17(2):214–234, 2001.

[42] Béatrice Fuchs, Jean Lieber, Alain Mille, and Amedeo Napoli. Towards a unified theory of adaptation in case-based reasoning. In *International Conference on Case-Based Reasoning*, pages 104–117. Springer, 1999.

[43] Lisa Cummins. *Combining and choosing case base maintenance algorithms*. PhD thesis, University College Cork, 2013.

[44] David B Leake and David C Wilson. Categorizing case-base maintenance: Dimensions and directions. In *European Workshop on Advances in Case-Based Reasoning*, pages 196–207. Springer, 1998.

[45] Delu Wang, Kaidi Wan, and Wenxiao Ma. Emergency decision-making model of environmental emergencies based on case-based reasoning method. *Journal of environmental management*, 262:110382, 2020.

[46] Alfonso González-Briones, Javier Prieto, Fernando De La Prieta, Enrique Herrera-Viedma, and Juan M Corchado. Energy optimization using a case-based reasoning strategy. *Sensors*, 18(3):865, 2018.

[47] I-Cheng Yeh and Tzu-Kuang Hsu. Building real estate valuation models with comparative approach through case-based reasoning. *Applied Soft Computing*, 65:260–271, 2018.

[48] A Rahman, C Slamet, W Darmalaksana, Y A Gerhana, and M A Ramdhani. Expert system for deciding a solution of mechanical failure in a car using case-based reasoning. *IOP Conference Series: Materials Science and Engineering*, 288:012011, Jan 2018.

[49] Eduardo Lupiani, Jose M. Juarez, Jose Palma, and Roque Marin. Monitoring elderly people at home with temporal case-based reasoning. *Knowledge-Based Systems*, 134:116–134, Oct 2017.

[50] Renata Saraiva, Mirko Perkusich, Lenardo Silva, Hyggo Almeida, Clauirton Siebra, and Angelo Perkusich. Early diagnosis of gastrointestinal cancer by using case-based and rule-based reasoning. *Expert Systems with Applications*, 61:192–202, Nov 2016.

[51] HJ Gómez-Vallejo, B Uriel-Latorre, M Sande-Meijide, B Villamarín-Bello, Reyes Pavón, F Fdez-Riverola, and Daniel Glez-Pena. A case-based reasoning system for aiding detection and classification of nosocomial infections. *Decision Support Systems*, 84:104–116, 2016.

[52] Fabio Sartori, Alice Mazzucchelli, and Angelo Di Gregorio. Bankruptcy forecasting using case-based reasoning: The creperie approach. *Expert Systems with Applications*, 64:400–411, Dec 2016.

[53] A. Sene, B. Kamsu-Foguem, and P. Rumeau. Telemedicine framework using case-based reasoning with evidences. *Computer Methods and Programs in Biomedicine*, 121(1):21–35, Aug 2015.

[54] Klaus-Dieter Althoff. Case-based reasoning for decision support and diagnostic problem solving: The inreca approach. 1995.

[55] Abir Smiti and Maha Nssibi. Case based reasoning framework for covid-19 diagnosis case based reasoning framework for covid-19 diagnosis. 2019.

[56] Souad GUESSOUM. *La prise en charge de l'incertain dans le système RespiDiag*. PhD thesis, Université de Annaba-Badji Mokhtar, 2014.

[57] Janet L Kolodner and Robert M Kolodner. Using experience in clinical problem solving: Introduction and framework. *IEEE Transactions on Systems, Man, and Cybernetics*, 17(3):420–431, 1987.

[58] E Ray Bareiss, Bruce W Porter, and Craig C Wier. Protos: An exemplar-based learning apprentice. In *Machine Learning*, pages 112–127. Elsevier, 1990.

[59] Carol Bradburn and John Zeleznikow. The application of case-based reasoning to the tasks of health care planning. In *European Workshop on Case-Based Reasoning*, pages 365–378. Springer, 1993.

[60] Elisha TO Opiyo. Case-based reasoning for expertise relocation in support of rural health workers in developing countries. In *International Conference on Case-Based Reasoning*, pages 77–87. Springer, 1995.

[61] Lothar Gierl, Mathias Bull, and Rainer Schmidt. Cbr in medicine. In *Case-Based Reasoning Technology*, pages 273–297. Springer, 1998.

[62] Jean Lieber and Benoît Bresson. Case-based reasoning for breast cancer treatment decision helping. In *European Workshop on Advances in Case-Based Reasoning*, pages 173–185. Springer, 2000.

[63] Ziad El Balaa, Anne Strauss, Philippe Uziel, Kerstin Maximini, and R Traphoner. Fm-ultranet: a decision support system using case-based reasoning, applied to ultrasonography. In *Workshop on CBR in the Health Sciences*, volume 37, pages 0–3. Springer-Verlag ICCBR'03, NTNU, Trondheim, Norway, 2003.

[64] Rainer Schmidt. Case-based reasoning in medicine especially an obituary on lothar gierl. In *Advanced Computational Intelligence Paradigms in Healthcare–1*, pages 63–87. Springer, 2007.

[65] Souvik Chakraborty, Chiranjit Pal, Shambo Chatterjee, Baisakhi Chakraborty, and Nabin Ghoshal. Knowledge-based system architecture on cbr for detection of cholera disease. In *Intelligent Computing and Applications*, pages 155–165. Springer, 2015.

[66] Dongxiao Gu, Changyong Liang, and Huimin Zhao. A case-based reasoning system based on weighted heterogeneous value distance metric for breast cancer diagnosis. *Artificial intelligence in medicine*, 77:31–47, 2017.

[67] Zuzana Tocimáková, Ján Paralič, Dominik Pella, et al. Case-based reasoning for support of the diagnostics of cardiovascular diseases. *Studies in health technology and informatics*, 270:537–541, 2020.

[68] Maria Salamó and Maite López-Sánchez. Adaptive case-based reasoning using retention and forgetting strategies. *Knowledge-Based Systems*, 24(2):230–247, 2011.

[69] Ioannis Iglezakis, Thomas Reinartz, and Thomas R Roth-Berghofer. Maintenance memories: beyond concepts and techniques for case base maintenance. In *European Conference on Case-Based Reasoning*, pages 227–241. Springer, 2004.

[70] Shaul Markovitch and Paul D Scott. The role of forgetting in learning. In *Machine Learning Proceedings 1988*, pages 459–465. Elsevier, 1988.

[71] Asma Chebli, Akila Djebbar, Hayet Farida Marouani, and Hakim Lounis. Case-base maintenance: an approach based on active semi-supervised learning. *International Journal of Pattern Recognition and Artificial Intelligence*, page S0218001421510113, May 2021.

[72] Barry Smyth and Mark T Keane. Remembering to forget. In *Proc. 14th IJCAI*, pages 377–382. Citeseer, 1995.

[73] D Randall Wilson and Tony R Martinez. Reduction techniques for instance-based learning algorithms. *Machine learning*, 38(3):257–286, 2000.

[74] Nariman Nakhjiri, Maria Salamó, and Miquel Sànchez-Marrè. Reputation-based maintenance in case-based reasoning. *Knowledge-Based Systems*, 193:105283, 2020.

[75] Abir Smiti and Zied Elouedi. Wcoid-dg: An approach for case base maintenance based on weighting, clustering, outliers, internal detection and dbsan-gmeans. *Journal of computer and system sciences*, 80(1):27–38, 2014.

[76] Barry Smyth. Case-base maintenance. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 507–516. Springer, 1998.

[77] David B Leake and David C Wilson. Remembering why to remember: Performance-guided case-base maintenance. In *European Workshop on Advances in Case-Based Reasoning*, pages 161–172. Springer, 2000.

[78] Qiang Yang and Jing Wu. Keep it simple: A case-base maintenance policy based on clustering and information theory. In *Conference of the Canadian Society for Computational Studies of Intelligence*, pages 102–114. Springer, 2000.

[79] Guoqing Cao, Simon Shiu, and Xizhao Wang. A fuzzy-rough approach for case base maintenance. In *International Conference on Case-Based Reasoning*, pages 118–130. Springer, 2001.

[80] Simon CK Shiu, Daniel S Yeung, Cai H Sun, and Xi Z Wang. Transferring case knowledge to adaptation knowledge: An approach for case-base maintenance. *Computational Intelligence*, 17(2):295–314, 2001.

[81] Barry Smyth and Elizabeth McKenna. Building compact competent case-bases. In *International Conference on Case-Based Reasoning*, pages 329–342. Springer, 1999.

[82] Qiang Yang and Jun Zhu. A case-addition policy for case-base maintenance. *Computational Intelligence*, 17(2):250–262, 2001.

[83] Kirsti Racine and Qiang Yang. On the consistency management of large case bases: the case for validation. In *To appear in AAAI Technical Report-Verification and Validation Workshop*, page 1. Citeseer, 1996.

[84] Brigitte Chebel-Morello, Mohamed Karim Haouchine, and Noureddine Zerhouni. Case-based maintenance: Structuring and incrementing the case base. *Knowledge-Based Systems*, 88:165–183, 2015.

[85] Javad Hamidzadeh, Reza Monsefi, and Hadi Sadoghi Yazdi. Irahc: instance reduction algorithm using hyperrectangle clustering. *Pattern Recognition*, 48(5):1878–1889, 2015.

[86] Peter Hart. The condensed nearest neighbor rule (corresp.). *IEEE transactions on information theory*, 14(3):515–516, 1968.

[87] Dennis L Wilson. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man, and Cybernetics*, (3):408–421, 1972.

[88] Ivan Tomek et al. An experiment with the edited nearest-nieghbor rule. 1976.

[89] Jun Zhu and Qiang Yang. Remembering to add: competence-preserving case-addition policies for case-base maintenance. In *IJCAI*, volume 99, pages 234–241, 1999.

[90] Barry Smyth and Elizabeth McKenna. Building compact competent case-bases. In *International Conference on Case-Based Reasoning*, pages 329–342. Springer, 1999.

[91] Sarah Jane Delany and Pádraig Cunningham. An analysis of case-base editing in a spam filtering system. In *European Conference on Case-Based Reasoning*, pages 128–141. Springer, 2004.

[92] Chien-Hsing Chou, Bo-Han Kuo, and Fu Chang. The generalized condensed nearest neighbor rule as a data reduction method. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 2, pages 556–559. IEEE, 2006.

[93] Susan Craw, Stewart Massie, and Nirmalie Wiratunga. Informed case base maintenance: A complexity profiling approach. In *AAAI*, pages 1618–1621, 2007.

[94] Sarah Jane Delany. The good, the bad and the incorrectly classified: Profiling cases for case-base editing. In *International Conference on Case-Based Reasoning*, pages 135–149. Springer, 2009.

[95] Aijun Yan, Limin Qian, and Chunxiao Zhang. Memory and forgetting: An improved dynamic maintenance method for case-based reasoning. *Information Sciences*, 287:50–60, 2014.

[96] David Leake and Brian Schack. Flexible feature deletion: compacting case bases by selectively compressing case contents. In *International Conference on Case-Based Reasoning*, pages 212–227. Springer, 2015.

[97] Amira Abdel-Aziz and Eyke Hüllermeier. Case base maintenance in preference-based cbr. In *International Conference on Case-Based Reasoning*, pages 1–14. Springer, 2015.

[98] Eduardo Lupiani, Stewart Massie, Susan Craw, Jose M Juarez, and Jose Palma. Case-base maintenance with multi-objective evolutionary algorithms. *Journal of intelligent information systems*, 46(2):259–284, 2016.

[99] Ditty Mathew and Sutanu Chakraborti. Competence guided model for case-base maintenance. In *IJCAI*, pages 4904–4908, 2017.

[100] Abir Smiti and Zied Elouedi. Scbm: soft case base maintenance method based on competence model. *Journal of Computational Science*, 25:221–227, 2018.

[101] Abir Smiti and Zied Elouedi. Dynamic maintenance case base using knowledge discovery techniques for case based reasoning systems. *Theoretical Computer Science*, 817:24–32, 2020.

[102] Mohssen Mohammed, Muhammad Badruddin Khan, and Eihab Bashier Mohammed Bashier. *Machine learning: algorithms and applications*. Crc Press, 2016.

[103] Burr Settles. Active learning literature survey. 2009.

[104] Nawel Zemmal. *Techniques d'apprentissage pour la selection de données: Application à la reconnaissance .des formes*. PhD thesis, Badji Mokhtar, 2017.

[105] Xindong Wu, Vipin Kumar, J Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J McLachlan, Angus Ng, Bing Liu, S Yu Philip, et al. Top 10 algorithms in data mining. *Knowledge and information systems*, 14(1):1–37, 2008.

[106] Abir Smiti. When machine learning meets medical world: Current status and future challenges. *Computer Science Review*, 37:100280, 2020.

[107] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

[108] A Agrawala. Learning with a probabilistic teacher. *IEEE Transactions on Information Theory*, 16(4):373–379, 1970.

[109] Xiaojin Jerry Zhu. Semi-supervised learning literature survey. 2005.

[110] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100, 1998.

[111] Kristin Bennett, Ayhan Demiriz, et al. Semi-supervised support vector machines. *Advances in Neural Information processing systems*, pages 368–374, 1999.

[112] Monica Bianchini, Marco Maggini, and Lakhmi C Jain. *Handbook on neural information processing*, volume 7. Springer, 2013.

[113] Razieh Sheikhpour, Mehdi Agha Sarram, Sajjad Gharaghani, and Mohammad Ali Zare Chahooki. A survey on semi-supervised feature selection methods. *Pattern Recognition*, 64:141–158, 2017.

[114] Dana Angluin. Queries and concept learning. *Machine learning*, 2(4):319–342, 1988.

[115] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996.

[116] Les E Atlas, David A Cohn, and Richard E Ladner. Training connectionist networks with queries and selective sampling. In *Advances in neural information processing systems*, pages 566–573. Citeseer, 1990.

[117] David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine learning*, 15(2):201–221, 1994.

[118] David D Lewis and William A Gale. A sequential algorithm for training text classifiers. In *SIGIR'94*, pages 3–12. Springer, 1994.

[119] Yi Wu, Igor Kozintsev, Jean-Yves Bouguet, and Carole Dulong. Sampling strategies for active learning in personal photo retrieval. In *2006 IEEE International Conference on Multimedia and Expo*, pages 529–532. IEEE, 2006.

[120] Ziang Liu and Dongrui Wu. Integrating informativeness, representativeness and diversity in pool-based sequential active learning for regression. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2020.

[121] Manal Al Ghamdi, Mingqi Li, Mohamed Abdel-Mottaleb, and Mohamed Abou Shousha. Semi-supervised transfer learning for convolutional neural networks for glaucoma detection. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3812–3816. IEEE, 2019.

[122] Wenjia Bai, Ozan Oktay, Matthew Sinclair, Hideaki Suzuki, Martin Rajchl, Giacomo Tarroni, Ben Glocker, Andrew King, Paul M Matthews, and Daniel Rueckert. Semi-supervised learning for network-based cardiac mr image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 253–260. Springer, 2017.

[123] Yingda Xia, Fengze Liu, Dong Yang, Jinzheng Cai, Lequan Yu, Zhuotun Zhu, Daguang Xu, Alan Yuille, and Holger Roth. 3d semi-supervised learning with uncertainty-aware multi-view co-training. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3646–3655, 2020.

[124] Daniel Chamberlain, Rahul Kodgule, Daniela Ganelin, Vivek Miglani, and Richard Ribón Fletcher. Application of semi-supervised deep learning to lung sound analysis. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 804–807. IEEE, 2016.

[125] Wenqing Sun, Tzu-Liang Bill Tseng, Jianying Zhang, and Wei Qian. Computerized breast cancer analysis system using three stage semi-supervised learning method. *Computer methods and programs in biomedicine*, 135:77–88, 2016.

[126] Chao Deng and M Zu Guo. A new co-training-style random forest for computer aided diagnosis. *Journal of Intelligent Information Systems*, 36(3):253–281, 2011.

[127] Mohammed El Amine Bechar, Nesma Settouti, Vincent Barra, and Mohamed Amine Chikh. Semi-supervised superpixel classification for medical images segmentation: application to detection of glaucoma disease. *Multidimensional Systems and Signal Processing*, 29(3):979–998, 2018.

[128] Nesma Settouti, Mostafa El Habib Daho, Mohammed El Amine Lazouni, and Mohammed Amine Chikh. Random forest in semi-supervised learning (co-forest). In *2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, pages 326–329. IEEE, 2013.

[129] Yang Liu, Zhian Xing, Chao Deng, Ping Li, and Maozu Guo. Automatically detecting lung nodules based on shape descriptor and semi-supervised learning. In *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, volume 1, pages V1–647. IEEE, 2010.

[130] Ming Li and Zhi-Hua Zhou. Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 37(6):1088–1098, 2007.

[131] Dwarikanath Mahapatra, Franciscus M Vos, and Joachim M Buhmann. Active learning based segmentation of crohns disease from abdominal mri. *Computer methods and programs in biomedicine*, 128:75–85, 2016.

[132] Nawel Zemmal, Nabiha Azizi, Nilanjan Dey, and Mokhtar Sellami. Adaptive semi supervised support vector machine semi supervised learning with features cooperation for breast cancer classification. *Journal of Medical Imaging and Health Informatics*, 6(1):53–62, 2016.

[133] Elaheh Moradi, Antonietta Pepe, Christian Gaser, Heikki Huttunen, Jussi Tohka, Alzheimer's Disease Neuroimaging Initiative, et al. Machine learning framework for early mri-based alzheimer's conversion prediction in mci subjects. *Neuroimage*, 104:398–412, 2015.

[134] Hala Helmi, Daphne Teck, Ching Lai, and Jonathan M Garibaldi. Semi-supervised techniques in breast cancer classification. In *12th Annual Workshop on Computational Intelligence (UKCI)*, 2012.

[135] Roman Filipovych, Christos Davatzikos, Alzheimer's Disease Neuroimaging Initiative, et al. Semi-supervised pattern classification of medical images: application to mild cognitive impairment (mci). *NeuroImage*, 55(3):1109–1119, 2011.

[136] Liansheng Wang, Shusheng Li, Yiping Chen, Jiankun Lin, Changhua Liu, Xiantong Zeng, and Shuo Li. Direct aneurysm volume estimation by multi-view semi-supervised manifold learning. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 1222–1225. IEEE, 2017.

[137] Zhengxia Wang, Xiaofeng Zhu, Ehsan Adeli, Yingying Zhu, Feiping Nie, Brent Munsell, Guorong Wu, et al. Multi-modal classification of neurodegenerative disease by progressive graph-based transductive learning. *Medical image analysis*, 39:218–230, 2017.

[138] Magnus Borga, Thord Andersson, and Olof Dahlqvist Leinhard. Semi-supervised learning of anatomical manifolds for atlas-based segmentation of medical images. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 3146–3149. IEEE, 2016.

[139] Le An, Ehsan Adeli, Mingxia Liu, Jun Zhang, and Dinggang Shen. Semi-supervised hierarchical multimodal feature and sample selection for alzheimer's disease diagnosis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 79–87. Springer, 2016.

[140] Rachel Sparks and Anant Madabhushi. Out-of-sample extrapolation utilizing semi-supervised manifold learning (ose-ssl): content based image retrieval for histopathology images. *Scientific reports*, 6(1):1–15, 2016.

[141] Wenqing Sun, Tzu-Liang Bill Tseng, Jianying Zhang, and Wei Qian. Enhancing deep convolutional neural network scheme for breast cancer diagnosis with unlabeled data. *Computerized Medical Imaging and Graphics*, 57:4–9, 2017.

[142] Xing Chen and Gui-Ying Yan. Semi-supervised learning for potential human microrna-disease associations inference. *Scientific reports*, 4(1):1–10, 2014.

[143] Anca Ciurte, Xavier Bresson, Olivier Cuisenaire, Nawal Houhou, Sergiu Nedevschi, Jean-Philippe Thiran, and Meritxell Bach Cuadra. Semi-supervised segmentation of ultrasound images based on patch representation and continuous min cut. *PloS one*, 9(7):e100972, 2014.

[144] Chihyun Park, Jaegyoon Ahn, Hyunjin Kim, and Sanghyun Park. Integrative gene network construction to analyze cancer recurrence using semi-supervised learning. *PloS one*, 9(1):e86309, 2014.

[145] Xiao Liu, Jun Shi, Shichong Zhou, and Minhua Lu. An iterated laplacian based semi-supervised dimensionality reduction for classification of breast cancer on ultrasound images. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4679–4682. IEEE, 2014.

[146] Yan Yan, Romer Rosales, Glenn Fung, and Jennifer Dy. Modeling multiple annotator expertise in the semi-supervised learning scenario. *arXiv preprint arXiv:1203.3529*, 2012.

[147] Kayhan N Batmanghelich, H Ye Dong, Kilian M Pohl, Ben Taskar, Christos Davatzikos, et al. Disease classification and prediction via semi-supervised dimensionality reduction. In *2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1086–1090. IEEE, 2011.

[148] Jun-Bao Li, Yang Yu, Zhi-Ming Yang, and Lin-Lin Tang. Breast tissue image classification based on semi-supervised locality discriminant projection with kernels. *Journal of medical systems*, 36(5):2779–2786, 2012.

[149] Mingguang Shi and Bing Zhang. Semi-supervised learning improves gene expression-based prediction of cancer recurrence. *Bioinformatics*, 27(21):3017–3023, 2011.

[150] Jiang Wu, Yuan-Bo Diao, Meng-Long Li, Ya-Ping Fang, and Dai-Chuan Ma. A semi-supervised learning based method: Laplacian support vector machine used in diabetes disease diagnosis. *Interdisciplinary Sciences: Computational Life Sciences*, 1(2):151–155, 2009.

[151] Wei Huang, Kap Luk Chan, Yan Gao, Jiayin Zhou, and Vincent Chong. Semi-supervised nasopharyngeal carcinoma lesion extraction from magnetic resonance images using online spectral clustering with a learned metric. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 51–58. Springer, 2008.

[152] Dwarikanath Mahapatra. Combining multiple expert annotations using semi-supervised learning and graph cuts for medical image segmentation. *Computer Vision and Image Understanding*, 151:114–123, 2016.

[153] Kedir M Adal, Désiré Sidibé, Sharib Ali, Edward Chaum, Thomas P Karnowski, and Fabrice Mériaudeau. Automated detection of microaneurysms using scale-adapted blob analysis and semi-supervised learning. *Computer methods and programs in biomedicine*, 114(1):1–10, 2014.

[154] E van Rikxoort, M Galperin-Aizenberg, J Goldin, TTJP Kockelkorn, B van Ginneken, and M Brown. Multi-classifier semi-supervised classification of tuberculosis patterns on chest ct scans. In *The Third International Workshop on Pulmonary Image Analysis*, pages 41–48, 2010.

[155] Xiaomeng Li, Lequan Yu, Hao Chen, Chi-Wing Fu, and Pheng-Ann Heng. Semi-supervised skin lesion segmentation via transformation consistent self-ensembling model. *arXiv preprint arXiv:1808.03887*, 2018.

[156] Mohammad Peikari, Sherine Salama, Sharon Nofech-Mozes, and Anne L Martel. A cluster-then-label semi-supervised learning approach for pathology image classification. *Scientific reports*, 8(1):1–13, 2018.

[157] Qiao Zheng, Hervé Delingette, and Nicholas Ayache. Explainable cardiac pathology classification on cine mri with motion characterization by semi-supervised learning of apparent flow. *Medical image analysis*, 56:80–95, 2019.

[158] Ammara Masood and Adel Al-Jumaily. Semi advised learning and classification algorithm for partially labeled skin cancer data analysis. In *2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)*, pages 1–4. IEEE, 2017.

[159] Zhi-Hua Zhou. When semi-supervised learning meets ensemble learning. *Frontiers of Electrical and Electronic Engineering in China*, 6(1):6–16, 2011.

[160] Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[161] Dekai Wu, Grace Ngai, and Marine Carpuat. A stacked, voted, stacked model for named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, pages 200–203, 2003.

[162] Ludmila I Kuncheva. *Combining pattern classifiers: methods and algorithms*. John Wiley & Sons, 2014.

[163] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, Aug 1996.

[164] Kevin Woods, W. Philip Kegelmeyer, and Kevin Bowyer. Combination of multiple classifiers using local accuracy estimates. *IEEE transactions on pattern analysis and machine intelligence*, 19(4):405–410, 1997.

[165] Shunkai Fu, Michel C Desmarais, and Fan Li. One-pass learning algorithm for fast recovery of bayesian network. In *FLAIRS Conference*, pages 53–58, 2008.

[166] Niels Landwehr, Mark Hall, and Eibe Frank. Logistic model trees. *Machine learning*, 59(1-2):161–205, 2005.

[167] William S Noble. What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567, 2006.

[168] John Platt. Sequential minimal optimization: A fast algorithm for training support vector machines. 1998.

[169] Dongrui Wu. Pool-based sequential active learning for regression. *IEEE transactions on neural networks and learning systems*, 30(5):1348–1359, 2018.

[170] Xueying Zhan and Antoni Bert Chan. Aldataset: a benchmark for pool-based active learning. *arXiv preprint arXiv:2010.08161*, 2020.

[171] Trupti M Kodinariya and Prashant R Makwana. Review on determining number of cluster in k-means clustering. *International Journal*, 1(6):90–95, 2013.

[172] Aristidis Likas, Nikos Vlassis, and Jakob J Verbeek. The global k-means clustering algorithm. *Pattern recognition*, 36(2):451–461, 2003.

[173] Janmenjoy Nayak, Bighnaraj Naik, and HSr Behera. Fuzzy c-means (fcm) clustering algorithm: a decade review from 2000 to 2014. *Computational intelligence in data mining-volume 2*, pages 133–149, 2015.

[174] James C Bezdek, Robert Ehrlich, and William Full. Fcm: The fuzzy c-means clustering algorithm. *Computers & geosciences*, 10(2-3):191–203, 1984.

[175] Mohiuddin Ahmed and Abdun Naser Mahmood. A novel approach for outlier detection and clustering improvement. In *2013 IEEE 8th Conference on Industrial Electronics and Applications (iciea)*, pages 577–582. IEEE, 2013.

[176] Pádraig Cunningham and Gabriele Zenobi. Case representation issues for case-based reasoning from ensemble research. In *International Conference on Case-Based Reasoning*, pages 146–157. Springer, 2001.

[177] Alican Dogan and Derya Birant. A weighted majority voting ensemble approach for classification. In *2019 4th International Conference on Computer Science and Engineering (UBMK)*, pages 1–6. IEEE, 2019.

[178] Ishan Misra, Abhinav Shrivastava, and Martial Hebert. Watch and learn: Semi-supervised learning for object detectors from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3593–3602, 2015.

[179] Jurica Levatić, Michelangelo Ceci, Dragi Kocev, and Sašo Džeroski. Self-training for multi-target regression with tree ensembles. *Knowledge-based systems*, 123:41–60, 2017.

[180] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[181] Miguel Patrício, José Pereira, Joana Crisóstomo, Paulo Matafome, Manuel Gomes, Raquel Seiça, and Francisco Caramelo. Using resistin, glucose, age and bmi to predict the presence of breast cancer. *BMC cancer*, 18(1):1–8, 2018.

[182] Davide Chicco and Giuseppe Jurman. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. *BMC medical informatics and decision making*, 20(1):16, 2020.

[183] I-Cheng Yeh, King-Jang Yang, and Tao-Ming Ting. Knowledge discovery on rfm model using bernoulli sequence. *Expert Systems with Applications*, 36(3):5866–5871, 2009.

[184] Matthias Elter, Rüdiger Schulz-Wendtland, and Thomas Wittenberg. The prediction of breast cancer biopsy outcomes using two cad approaches that both emphasize an intelligible decision process. *Medical physics*, 34(11):4164–4172, 2007.

[185] Ahlem Refai, Hayet Farida Merouani, and Hayet Aouras. Maintenance of a bayesian network: application using medical diagnosis. *Evolving Systems*, 7(3):187–196, 2016.

[186] Mustafa Kemal Yöntem, ADEM Kemal, Tahsin Ilhan, and Serhat KILIÇARSLAN. Divorce prediction using correlation based feature selection and artificial neural networks. *Nevşehir Hacı Bektaş Veli Üniversitesi SBE Dergisi*, 9(1):259–273, 2019.

[187] Yogish Sabharwal, Nishant Sharma, and Sandeep Sen. Nearest neighbors search using point location in balls with applications to approximate voronoi decompositions. *Journal of Computer and System Sciences*, 72(6):955–977, 2006.

[188] David W Aha, Dennis Kibler, and Marc K Albert. Instance-based learning algorithms. *Machine learning*, 6(1):37–66, 1991.