AUTOMATED META-ACTIONS DISCOVERY FOR PERSONALIZED MEDICAL TREATMENTS

by

Hakim Touati

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Computing and Information Systems

Charlotte

2014

Approved by:

____

Dr. Zbigniew W. Raś

____

Dr. James Studnicki

____

Dr. Wlodek Zadrozny

____

Dr. Jing Yang

____

Dr. Mohammad Kazemi

ABSTRACT

HAKIM TOUATI. Automated meta-actions discovery for personalized medical treatments. (Under the direction of DR. ZBIGNIEW W. RAŚ)

Healthcare, among other domains, provides an attractive ground of work for knowledge discovery researchers. There exist several branches of health informatics and health data-mining from which we find actionable knowledge discovery is underserved. Actionable knowledge is best represented by patterns of structured actions that inform decision makers about actions to take rather than providing static information that may or may not hint to actions. The Action rules model is a good example of active structured action patterns that informs us about the actions to perform to reach a desired outcome. It is augmented by the meta-actions model that represents passive structured effects triggered by the application of an action. In this dissertation, we focus primarily on the meta-actions model that can be mapped to medical treatments and their effects in the healthcare arena. Our core contribution lies in structuring meta-actions and their effects (positive, neutral, negative, and side effects) along with mining techniques and evaluation metrics for meta-action effects. In addition to the mining techniques for treatment effects, this dissertation provides analysis and prediction of side effects, personalized action rules, alternatives for treatments with negative outcomes, evaluation for treatments success, and personalized recommendations for treatments. We used the tinnitus handicap dataset and the Healthcare Cost and Utilization Project (HCUP) Florida State Inpatient Databases (SID 2010) to validate our work. The results show the efficiency of our methods.

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Zbigniew W. Raś, for his direction and mentoring in my PhD studies. I not only value Dr. Raś as a great scientist, but also as an inspiring and kind person. He is one of the most knowledgable and humble people I have had the chance to meet. "Thank you Dr. Raś. I am forever grateful for the training, support, and mentoring you have provided me with throughout my studies, and the confidence you have given me in research."

I would also like to thank all the faculty members for helping me accomplish my goals and continuously supporting me. Particularly, Dr. Mohamed Shehab for his research training and support, Dr. William Tolone for his teaching training and support, and Dr. James Studnicki for his generous donation of datasets and collaboration.

I am grateful for all of my colleagues and friends who have offered their help. I would like to particularly thank, Fadi Fuad Mohsen, Saeed Al-Haj, and Ayman Hajja for evaluating my articles, reviewing my dissertation, and providing me with unremitting support.

Thank you all for your unceasing support and encouragement.

DEDICATION

I would like to dedicate my dissertation to my beloved parents, Abdelkrim Touati and Nadjia Keltouma Touati, born Arezki, and my brothers and sister. ”Mother and Father, I am truly blessed and honored to have your support in life. Thank you for providing the path for me to become the person I was intended to become. Your unconditional love and support have enabled me to continuously develop as a scholar.”

I also would like to dedicate this dissertation to my uncle, Dr. Abderahmane Touati, and his family. I will never forget their support and encouragement throughout my life in the United States.

TABLE OF CONTENTS

LIST OF FIGURES ix

LIST OF TABLES x

CHAPTER 1: INTRODUCTION 1

1.1 Motivation 1

1.2 Research Problem 6

1.3 State of the Art 7

CHAPTER 2: PRELIMINARIES 10

2.1 Static Model 10

2.2 Association Rules 12

2.3 Action Model 13

2.4 Action Rules 14

CHAPTER 3: DATASET AND SOLUTION OVERVIEW 18

3.1 Tinnitus Dataset Description 18

3.2 Completed Work Using the Tinnitus Dataset 20

3.3 HCUP Dataset Description 22

3.4 Completed Work Using HCUP Dataset 23

CHAPTER 4: META ACTIONS 25

4.1 Meta-actions Overview 26

4.2 Meta-actions Based on Action Terms 29

4.2.1 Meta-actions Representation by Influence Matrices 29

4.2.2 Meta-actions Mining 30

4.2.3 Meta-actions and Action Terms Evaluation 34

4.2.4 Tinnitus Meta-actions Mining 35

4.3 Meta-actions Based on Action Sets 38

4.3.1 Meta-actions Representation by an Ontology 39

4.3.2 Meta-action Extraction 39

4.3.3 Meta-actions and Action Sets Evaluation 42

4.3.4 Surgical Meta-actions Mining for HCUP 44

4.4 Chapter Conclusion 46

CHAPTER 5: SIDE-EFFECTS ANALYSIS AND THEIR USABILITY 47

5.1 Side Effects Based on Action Terms 48

5.1.1 Meta-actions Side Effects 48

5.1.2 Action Rules Side Effects 49

5.2 Side Effects Based on Action Sets 49

5.2.1 Negative Action Sets Mining 50

5.2.2 Patients' Clustering Based on Side Effects 51

5.2.3 Negative Side Effects Evaluation 52

5.2.4 Negative Side Effects Predictions 53

5.2.5 Experimental Evaluation 55

5.3 Personalized Action Rules Extraction Based on Object Clustering 61

5.3.1 Personalized Object Grouping 61

5.3.2 Experiments 67

5.4 Chapter Conclusion 76

CHAPTER 6: HEALTH CARE META-ACTIONS IN PLAY 78

6.1 Personalized Action Rules Reduction 79

6.1.1 Action Rules Substitution 79

6.1.2 Experiments and Evaluation 84

6.2 Meta-action as a Tool for Action Rules Evaluation 89

6.2.1 Meta-Actions Versus Action Rules 89

6.2.2 Experiments 96

6.3 Chapter Conclusion 103

CHAPTER 7: CONCLUSION AND DISCUSSION 104

REFERENCES 106

LIST OF FIGURES

FIGURE 1: Meta-actions mining methodology. 31

FIGURE 2: Ontology representation of a meta-action. 39

FIGURE 3: Meta-action confidence for surgical treatments. 45

FIGURE 4: Ontology representation of a meta-action with negative effects. 51

FIGURE 5: Negative action sets confidence proportion. 56

FIGURE 6: Negative action sets confidence by size of clusters. 60

FIGURE 7: Three dimensional representation of object grouping. 62

FIGURE 8: Patient population by side effect groups. 73

FIGURE 9: Support of each grouping scheme. 74

FIGURE 10: Confidence of each grouping scheme. 75

FIGURE 11: Action rules reduction methodology. 80

FIGURE 12: Comparison of the execution measures for both SG and HA. 87

LIST OF TABLES

TABLE 1: Information system example. 11

TABLE 2: Mapping between attributes and concepts features. 23

TABLE 3: Meta-actions influence matrix. 30

TABLE 4: Tinnitus meta-actions evaluation. 36

TABLE 5: Action sets confidence and likelihood. 44

TABLE 6: Negative clusters analysis for all meta-actions. 55

TABLE 7: Negative clusters analysis for meta-action 34. 57

TABLE 8: Negative clusters analysis for meta-action 44. 57

TABLE 9: Examples of negative action sets for all meta-actions. 58

TABLE 10: Negative clusters analysis for meta-action 43. 59

TABLE 11: Negative clusters analysis for meta-action 45. 59

TABLE 12: Number of patients by side effects groups. 67

TABLE 13: Action rules by side effects groups. 68

TABLE 14: Action rules by antecedent side grouping. 70

TABLE 15: Number of patients by side effects and action rules grouping. 71

TABLE 16: Number of patients by meta-action. 72

TABLE 17: Action rules by meta-actions groups. 72

TABLE 18: Number of patients by meta-actions and side effects grouping. 73

TABLE 19: Tinnitus action rules reduction statistics. 86

TABLE 20: Meta-actions vs. action rules. 92

TABLE 21: Description of CCS single level procedure codes. 96

TABLE 22: Description of CCS single level diagnosis codes [4]. 97

TABLE 23: Influence matrix like table reporting support and confidence. 100

CHAPTER 1: INTRODUCTION

1.1 Motivation

Knowledge Discovery, also referred to as data-mining in the business world, is the process of searching large collections of structured or unstructured data for the purpose of extracting useful knowledge. This Knowledge is commonly represented by data patterns along with their properties derived from the data searching process. The extracted patterns represent structured associations and correlations among data items that can be interpreted by decision makers as useful information that allow them to take actions. Data-mining is used in several domains, such as banking, distribution, and healthcare. In healthcare informatics, data-mining is mainly used for the evaluation of the current implemented healthcare system and its providers. Researchers focus on the healthcare system for the purposes of reducing cost and providing better hospital logistics and insurance coverage. However, a handful of new companies and researchers are focusing on improving patients' care by providing better personalized treatments. For instance, Flatiron [15] is a business and clinical intelligence Oncology Cloud platform for cancer care providers that provides actionable insights based on their oncology data. Another good example is AudaxHealth [14], which offers a targeted social network called Zensey for healthcare players and takes a proactive strategy to engage people in their health early on, before requiring care, by providing useful recommendations.

Improving healthcare is a very challenging task due to the privacy policies implemented by HIPPAA [19] resulting in the lack of publicly available datasets. In this dissertation, we aim at analyzing personalized treatment patterns and their underlying side effects by answering a number of questions: how actionable treatment patterns should be structured, how data structures of actionable treatment patterns are mined efficiently, how to present lower costs personalized treatment recommendations for patients, how possible side effects are identified and avoided, how to cluster patients for personalized treatments based on side effects, how to present alternative treatments for severe cases, and finally how to evaluate and optimize actionable tasks. To answer these questions, we use state of the art research, such as supervised and unsupervised learning, collaborative filtering, and action rules. Moreover, we develop new methods of information retrieval and actionable knowledge.

Our main model in this dissertation revolves around structuring the action / reaction to extract actionable data. In other words, we model how we want our actionable answers (or patterns) to be structured. In healthcare, we partition actions into two different types. The first type models an action that is yet to happen with its desired known reactions; for example, a doctor that might know what diagnoses he wants to cure but does not know the treatment to prescribe yet. In this example, diagnoses being cured are reactions of actions that have not been taken yet. The second type is based on modeling an action that already took place and triggers (or results in) certain reactions that have yet to be extracted; for example, when a doctor takes the action of prescribing a treatment for a patient, we view the treatment as the action that is being taken while we view the treatment effects as the reactions triggered that are still unknown.

There exist a known model called action rules that structures actions of the first type into rules. Action rules that were first introduced in [25] observe patterns, recorded on an information system by domain experts applying their domain knowledge and expertise in real world situations. They provide efficient solutions to help naive system users solve real world problems. Action rule discovery algorithms are widely recognized and used in multiple fields. They provide a way of discovering the best actions to perform to reach a more profitable state. This is done by modeling the correlation between the values' transitions of the objects' properties and the desired decision property values' transition. The object properties correlation result in a cascading effect that ultimately triggers a change in the decision value (or outcome).

There has been an increasing interest on action rule discovery algorithms since their introduction by Raś and Wieczorkowska in [25]. Action rules have been used in healthcare to understand experts' practices and improve patients care [24, 38, 40]. They are also used in distribution and customer loyalty systems, and also applicable in a wide range of industries such as education and banking.

Meta-actions are the triggers to those action rules; they provide a tool to control action rules execution. Meta-actions are a good example of the second type of actions that trigger specific reaction called meta-actions' effects. They are commonly represented by an influence matrix containing a set of feature values transitions that they trigger; however, they are also represented by an ontology when multivalued features are used. Meta-actions are typically defined by domain knowledge and by domain experts, and used by system users. When meta-actions are used to trigger an action rule, they trigger changes in objects state within the system to execute the action rule. However, they might provoke changes in objects features that will not only trigger the action rules targeted, but can also cause side effects that could be negative. The negative side effects can negatively affect (or damage) the objects features outside of the executed action rule scope. Naive system users might not know about those negative side effects; thus, they will not be taken into consideration even though they might be harmful.

In addition, it is not always the case that meta-actions are provided and available in a ready-to-use format; therefore, meta-actions often have to be mined and properly formatted. Meta-actions can be formatted into two types of data structures: action terms and action sets. Both representations capture the data actionability and the reactions of objects' features. Furthermore, action rules are commonly discovered and then chosen independently of the meta-actions and their negative side effects. However, action rules and their triggering meta-actions are interrelated, and they should be discovered together along with taking into consideration the objects and their related negative side-effects. Since the discovery process is bounded by the system in use, it is important to integrate the action rules and meta-actions discovery.

In the remainder of this chapter, we will define the problem studied and state our hypothesis, and we will briefly discuss the state of the art of current research. In Chapter 2, we will define several concepts that will constitute a big part of the language and technical jargon used throughout this dissertation. We will then give a quick overview of the work completed and a description of the datasets used to complete this dissertation in Chapter 3. Meta-action is the core model used in this dissertation. A detailed description and definition of meta-actions is given in Chapter 4 along with examples illustrations. In addition, the representation models, mining techniques, and evaluation metrics along with a real world dataset mining demonstration for meta-actions is presented in Chapter 4. After the presentation of our core model in the meta-actions chapter, we undergo a detailed side effects study and analysis in Chapter 5; this chapter describes the different types of side effects and their representations. Furthermore, patients' grouping schemes based on predefined side-effects are discussed and analyzed, and patients clustering scheme based on extracted side-effects is described and analyzed along with a personalized patients side effects prediction based on treatments. In Chapter 6, we present few projects that describe the usability of meta-actions and their side-effects. We first start by describing how we can reduce or substitute action rules to improve their execution confidence and reduce their side-effects on patients. We then compare meta-actions with action rules and describe how they can be used to evaluate action rules. The last topic of this chapter describes the possibility of using meta-actions to recommend personalized treatments for patients. Finally, we conclude in Chapter 7, by summarizing our contribution in the field of actionable knowledge discovery and state the shortcomings of the techniques presented in this dissertation.

1.2 Research Problem

The problem that we address in this dissertation concerns the structure of actionable knowledge in terms of meta-actions and their effects, along with providing meta-actions mining and representation techniques. The problem of mining meta-actions was not studied before and was left to domain experts. In addition, objects display different behavior for different meta-actions; therefore, there is a need for personalized action rules extraction based on meta-actions side effects. Furthermore, meta-actions application introduces negative side effects in some scenarios; hence, the need for a thorough side effects analysis is critical. In fact, the meta-actions used to trigger the execution of action rules introduces negative effects. For example, given a bank customer that is 24 years old, has medium salary, medium monthly expenses, high savings, low interest rate, and average loan profitability; if we apply meta-actions to increase the interest rate which would trigger an action rule increasing the loan profitability, we may as well trigger a decrease in customers savings, thus affecting the saving account profitability rather negatively. This scenario may not be suitable for the bank decision maker. Other important problems that we visited in this dissertation concern the medical treatments recommendations in healthcare and the negative side effects prediction.

Problem Statement: *We investigate the problem of structuring actionable patterns under the form of meta-actions and mining their effects in order to propose methods to minimize negative side effects for personalized recommendations.*

Hypothesis Statement: *We believe that meta-actions effects can be mined from previous object's behavior in large populations and that personalized action rules can be extracted based on objects negative side effects (reactions) to meta-actions.*

1.3 State of the Art

In this section, we give a brief background on the work that has been done in the areas prominent to this dissertation.

*Action Rules: There have been several research efforts on Action rules since their introduction by Raś and Wieczorkowska in [25]. The first effort to mine action rules from scratch was done in [12]. After their introduction, multiple action rules discovery techniques were presented [21, 28, 37, 39]. Actionable patterns were also discussed in [35]. However, they neither studied the execution of the action rules nor the triggers to their execution.

Action rules can also be seen as a composition of two classification rules as described in [23], where the authors described how to compute the support and confidence of an action rule based on its two composing classification rules. Action rules are already adopted by the banking industry, distribution industry, and healthcare industry to discover the right actions to follow in order to increase their profit . They have been investigated in [10, 12, 23, 2]. For instance, Wasyluk et al. [38] and Zhang et al. [40] studied action rules in the healthcare domain to improve patients care. There are a number of software packages available for discovering action rules. For instance, Action4ft-Miner module of the Lisp-Miner project developed by Jan Rauch's group discovers action rules under different constraints, which can be extracted based for the antecedent part of the rule [27].

*Meta-action: In action rules, neither the execution confidence nor the likelihood of the rules based on meta-actions were studied. Meta-actions were first introduced in

[37] as a higher level concepts used in modeling certain generalizations of action rules. They were described either by an influence matrix or an ontology as a set of value transitions in flexible attributes. Meta-actions were formalized in [22], and used in a pruning process with tree classifiers to discover action rules. In [36], the authors show the cascading effect of meta-actions leading to desired effects when generating association action rules and action paths. This work is slightly similar in the way that we use the cascading effect of meta-action atomic terms to model the correlation with the action rule.

The previous work on meta-actions neither studied the side effects of meta-actions nor the uncovered set substitutions. In this work we present, a meta-action mining, and action rules reduction mechanism based on meta-actions and negative side effects to improve the executability of the rules. We also present a personalized action rules extraction mechanism based on object grouping and negative side effects. To our knowledge, this is the first work that utilizes the advantage of meta-actions to improve the likelihood and confidence of execution of the action rules.

*Treatment Patterns: There has been several work for treatment patterns recognition. Kathryn et. al. [17] and Ramey et. al. [5] explored treatment patterns for a specific demographic population along with treatment outcomes. Lorigan et. al. [18] also explored treatment patterns and outcomes for patients with metastatic melanoma in the U.K. However, they did not study treatment effects patterns in terms of results. In [31], the authors mine medical articles for the disease and treatments as well as underlying side effects. This article is considers all three dimensions of treatments, side effects, and disease; however it fails in studying the results from real treatments prescribed to patients.

*Collaborative Filtering and Recommender Systems: There have been several contributions in the area of collaborative filtering and recommender systems. One of the first recommender systems is *Tapestry* [7] that cited the phrase of ”collaborative filtering (CF)”. In recommender systems, collaboration is not necessarily explicit as pointed in [29]. The assumption behind CF is that if two users rate several items similarly, they will rate other items similarly [8]. This way personalized recommendation can be derived.

CHAPTER 2: PRELIMINARIES

In this chapter, we present few concepts related to the static and action representations of data, along with rule based knowledge discovery. We first introduce the static model and association rules extracted from information systems, then we augment the definition of an information system to a decision system and introduce the concepts related to the action model and action rules. Furthermore, we will briefly visit the evaluation of association rules and action rules, and provide few examples. The concepts and language defined in this chapter will be used throughout the dissertation.

2.1 Static Model

In this section, we give a brief description of how static data is represented and stored in information systems. We also describe the state of an object in the context of information systems.

*Definition 1 (Information System) *By information system [20] we mean a triple of the form $S = (X, F, V)$ where:*

1. *X is a nonempty, finite set of objects.*
2. *F is a nonempty, finite set of features of the form $f : X \to 2^{V_f}$, which is a function for any $f \in F$, where V_f is called the domain of f.*

3. V is a finite set of attribute values such as: $V = \bigcup\{V_f : f \in F\}$.

If $f(x)$ is a singleton set, then $f(x)$ is written without parentheses (for instance, $\{v\}$ will be replaced by v). Table 1 represents an information system S with a set of objects $X = \{x_1, x_2, x_3, x_4, x_5\}$, a set of features $F = \{a, b, c, d\}$, and a set of feature values $V = \{a_1, a_2, b_1, b_2, b_3, c_1, c_2, c_3, d_1, d_2\}$.

Table 1: Information system example.

	a	b	c	d
x_1	a_1	b_2	c_2	d_1
x_2	a_2	b_2	c_2	d_2
x_3	a_2	b_1	c_3	d_1
x_4	a_1	b_3	c_1	d_2
x_5	a_2	b_1	c_1	d_1

In practice, data is not necessarily organized as we showed in the previous example; also information systems may not only have multivalued features, but also missing data and/or variable number of features.

To simplify the concept of objects with variable number of features, and the concept of features taking several values at the same time (multivalued features), we introduce the notion of *object state* and we define it as follows:

*Definition 2 (Object State) *The state of an object $x \in X$ in an information system S is defined by the set of features F_x that the object x is characterized by, and their respective values $F(x) = \bigcup\{f(x) : f \in F_x\}$*

For example, let us take the information system S represented by Table 1. The state of object $x_1 \in X$ is represented by $F(x_1) = \{a_1, b_2, c_2, d_1\}$.

2.2 Association Rules

Association rules are the representations of frequent patterns in transactional data that imply another associated pattern. They model frequent items in Market Basket Analysis (MBA). For instance, customers that often buy milk, bread and coffee in their basket will imply that milk, bread and coffee are associated, and form a frequent item. There have been several association rule mining algorithms such as [13, 30, 16, 1]. Let us have an information system S with a set of features F, and their respective values V called literals. The set $F(x_i) = \{f(x_i) : f \in F\}$ for the object $x_i \in X$, can be seen as a transaction t_i in the set of transactions T. We say that $A \rightarrow B$ is an association rule if:

- A is a subset of $F(x_i)$ for the object x_i.
- B is a subset of $F(x_i)$ for the object x_i.
- $A \cap B = \emptyset$.

An association rule is evaluated by its support $Sup(A \rightarrow B)$, which is the number of occurrence of $A \cup B$ in the transaction set T, or the percentage of occurrence of $A \cup B$ in the transaction set T. It is also evaluated by its confidence $Conf(A \rightarrow B)$, which is the percentage of occurrence of $A \cup B$ in the set of transactions where A only occurs. Formally, the support and confidence of a rule $A \rightarrow B$ are defined as follow:

- $Sup(A \rightarrow B) = |A \cup B|,\ \textit{where literals in } A \cup B \subseteq F(x_i) \textit{ are listed in } t_i.$
- $Conf(A \rightarrow B) = \frac{|A \cup B|}{|A|},\ \textit{where literals in } A \cup B \subseteq F(x_i) \textit{ are listed in } t_i.$

There have been few algorithms developed for frequent item-sets and association rules mining such as Apriori and its variations [1].

2.3 Action Model

In this section, we give a brief overview of action rules and some of the concepts used in the action model. Action rules take place in a decision system of the form $S = (X, F \cup \{d\}, V)$ that was introduced by Z. Pawlak in [20] and defined as follows:

*Definition 3 (Decision System) *By a decision system we mean* $S = (X, F \cup \{d\}, V)$*, where:*

1. *X is a set of objects, F is a set of classification features, d is a decision feature.*

2. *$f : X \to V_f$ is a function for any $f \in F$, where V_f is called the domain of f.*

3. *$d : X \to V_d$ is a function, where V_d is called the domain of d.*

4. *$V = V_F \cup V_d$, where $V_F = \bigcup\{V_f : f \in F\}$.*

Also, for each $x \in X$ and $f \in F$, we assume that value $f(x) \in V_f$ is classified as either positive (normal) or negative (abnormal). To be more precise, we assume that $F(x)$ denotes the set $\{f(x) : f \in F\}$, and that $F(x) = E_n(x) \cup E_p(x)$, where $E_p(x)$ is a set of positive values and $E_n(x)$ is a set of negative values for $x \in X$. If $f(x) \in E_n(x)$, then the value $f(x)$ is interpreted as abnormal (e.g. high temperature, cough, headache,). If $f(x) \in E_p(x)$, then value $f(x)$ is interpreted as a normal value for the feature f.

2.4 Action Rules

Action rules are rules that provide a set of actionable patterns to follow in order to transition the objects population from a certain state to a more profitable state with respect to a decision feature. They allow users to understand the correlations between transition patterns in the decision system, and construct actionable tasks that lead to a desirable outcome. In addition, action rules are composed of a decision feature d and classification features that are in turn divided into two sets: stable features F_{st}, and flexible features F_{fl} such that $F = F_{fl} \cup F_{st}$. Stable and Flexible features are defined next:

*Definition 4 (Stable Features) *Stable features are object properties that we do not have control over in the context of an information system. In other words, actions recommending changes of these features will fail.*

For example, birth date is a stable feature. This type of features is not used to model actions or transitions since their values do not change. They are commonly used to cluster the dataset.

*Definition 5 (Flexible features) *Flexible features are object properties that can transition from one value to another triggering a change in the object state.*

For instance, salary and benefits are flexible features since their values can change. Flexible features are the only possible features that can inform us about the possible changes an object may go through. However, to model possible actions, feature values transition, we need another concept which is defined as:

*Definition 6 (Atomic action term in S) *also called elementary action term in S, is an expression that defines a change of state for a distinct feature in S.*

For example, $(f, v_1 \to v_2)$ is an atomic action term which defines a change of value for attribute f in S from v_1 to v_2, where $v_1, v_2 \in V_f$. In the case when there is no change, we omit the right arrow sign; so for example, (f, v_1) means that the value of attribute f in S remains v_1, where $v_1 \in V_f$.

Atomic action terms model a single feature values transition pattern, but they do not model the association between feature values transition patterns. We augment the definition of atomic action terms to action terms by associating several transitions of feature values.

*Definition 7 (Action terms) *are defined as the smallest collection of expressions for a decision system S, such that:*

- *If t is an atomic action term in S, then t is an action term in S.*
- *If t_1, t_2 are action terms in S and $\wedge$ is a 2-argument functor called composition, then $t_1 \wedge t_2$ is a candidate action term in S.*
- *If t is a candidate action term in S and for any two atomic action terms $(f, v_1 \to v_2), (g, w_1 \to w_2)$ contained in t we have $f \neq g$, then t is an action term in S.*

Assuming that S is given, we will use from now on, *action term* instead of *action term in S*.

It is often important to have a differentiating factor between different action terms with regards to their usability. The domain of an action term helps identify them when needed and is defined as follow:

*Definition 8 (Domain of an action term) *The domain $Dom(t)$ of an action term t is the set of features values listed in the atomic action terms contained in t.*

For example, $t = [(f, v_1 \rightarrow v_2) \wedge (g, w_1)]$ is an action term that consists of two atomic action terms, namely $(f, v_1 \rightarrow v_2)$ and (g, w_1). Therefore, $Dom(t) = \{f, g\}$.

Action rules are expressions that take the following form: $r = [t_1 \Rightarrow t_2]$, where t_1, t_2 are action terms. The interpretation of the action rule r is that by triggering the action term t_1, we would get, as a result, the changes of states in action term t_2. We also assume that $Dom(t_1) \cup Dom(t_2) \subseteq F$, and $Dom(t_1) \cap Dom(t_2) = \emptyset$.

For example, $r = [[(f, v_1 \rightarrow v_2) \wedge (g, w_2)] \Rightarrow (d, d_1 \rightarrow d_2)]$ means that by changing the state of feature f from v_1 to v_2, and by keeping the state of feature g as w_2, we would observe a change in attribute d from the state d_1 to d_2, where d is commonly referred to as the decision attribute. In [23] it was observed that each action rule can be seen as a composition of two classification rules. For instance, the rule $r = [[(f, v_1 \rightarrow v_2) \wedge (g, w_2)] \Rightarrow (d, d_1 \rightarrow d_2)]$ is a composition of $r_1 = [(f, v_1) \wedge (g, w_2)] \rightarrow (d, d_1)$ and $r_2 = [(f, v_2) \wedge (g, w_2)] \rightarrow (d, d_2)$. This fact can be recorded by the equation $r = r(r_1, r_2)$. Also, the definition of support (Sup) and confidence ($Conf$) of an action rule is based on support and confidence of the classification rules (see below). Assume that action rule r is a composition of two classification rules r_1 and r_2. Then [23]:

- $Sup(r) = min\{card(sup(r_1)), card(sup(r_2))\}$,
- $Conf(r) = conf(r_1) \cdot conf(r_2)$.

The support of a classification rule or association rule can also be defined by the set of objects affected by the rule rather than their number. By the *support* of a classification rule $r = [[(f_1, f_{11}) \wedge (f_2, f_{21}) \wedge (f_3, f_{31}) \wedge \cdots \wedge (f_k, f_{k1})] \to (d, d_1)]$ in a decision system $S = (X, F \cup \{d\}, V)$, where $(\forall i \leq k)(f_i \in F \;\&\; f_{i1} \in V), d_1 \in V_d$, we mean the set $sup(r) = \{x \in X : (\forall i \leq k)[f_i(x) = f_{i1}] \;\&\; d(x) = d_1\}$, which represents the set of all the objects affected by the association.

CHAPTER 3: DATASET AND SOLUTION OVERVIEW

This section presents the datasets used to demonstrate the efficacy of our work in this dissertation. In the next three chapters, we present the work completed using the following two datasets and the experimental results.

3.1 Tinnitus Dataset Description

The first dataset used in this dissertation's experiments is obtained from the Tinnitus Handicap Inventory. It represents physician's observations on patients. The data contains three categories of observations on patients properties that are affected by tinnitus, which are: functioning (F), emotions (E), and how catastrophic it is (C). Each category consists of several related questions describing the patients state. There are 25 multiple choice questions altogether, and the answers to all of them can be mapped to numeric scores: Yes is 4, Sometimes is 2 and No is 0. To evaluate the overall status for each patient, physicians observed three features ScF, ScE, and ScC, which are the total score of functioning category, the total score of emotions category and the total score of catastrophic category, respectively. Those three scores represent the sum of all the answers scores for each category. Then feature ScT (total score) is generated by adding results of ScF, ScE, and ScC together to measure the tinnitus severity. The Tinnitus Handicap Inventory is completed during each patient's visit and stored with patient's ID, visit date and number, and patient's gender (g). Another aspect of the data was the treatment performed on the patients at each visit. The treatments performed on the patients at each visit were divided into four treatments that are: Hearing Aid (HA), Sound Generator (SG), Combination of HA and SG noted (CO), and a regular consultation (RC).

To be able to use the data in our experiments we had to perform a cleaning step along with a discretization step. In fact, the total number of patients visits is 2591 visit instances; however, there are multiple missing values and incomplete visit instances that had to be removed in order to be able to complete our experiments with the most refined data possible. After cleaning the data, we ended up having only 517 visit instances. We assumed that the classification features are the functioning (F), emotions (E), and catastrophic (C) features, and we kept their score values as they were already discretized. We further assumed that the side-effects were the three scores of each category of the features Sc F, Sc E, and Sc C. We discretized the side effects based on the improvements on the category score (score=1: positive side effects, which means that the score decreases) and the declining of the category score (score= 0: negative side effects which means that the score increases). The decision feature is the total score, and the main goal of the treatments was to decrease the total score. We also discretized the decision feature, the total score, based on its improvement and its declining (score=1 if improvement i.e. score decreases) and (score=0 if declining i.e. score increases).

3.2 Completed Work Using the Tinnitus Dataset

Improving the condition of patients affected by the tinnitus handicap is a tedious task that requires a deep analysis. The main challenges of the problem studied and work completed are presented as follow:

- Mining meta-actions based on action terms: The treatments introduced by experts in treating the tinnitus data are not always obvious and documented. For this reason, mining the treatments produced is a necessary task that requires analyzing the datasets at hand. In this work, we clustered the transactions in our dataset by patients and introduced a temporal order relationship between the patient's visits to mine patient's properties transitions. We also introduced two evaluation measures for meta-actions. Those measures are meta-action confidence and action term confidence and support.

- Action rules substitution: Extracted action rules are executed using meta-actions that introduce several negative side effects, and do not always result in desired outcomes. To ensure complete execution of action rules and decrease the negative side effects, we studied the possibility of substituting action rules by other action rules that appear to be less attractive and reduce original rules antecedent side. To evaluate the execution of the rules by meta-actions, we introduced the execution confidence metric that represents the confidence of executing a rule using a given set of meta-actions. Another useful metric introduced in this chapter is the utility that is used to compare between the execution confidences of the different rules and also the number of negative side effects introduced by the different respective meta-actions used for each rule.

- Personalized action rules based on patient grouping: In another effort to decrease the negative side effects, we aimed at extracting personalized action rules based on negative side effects. In fact, we believe that not all patients have the same negative side effects; thus, we should extract personalized action rules and treatments. In this work, we used three grouping schemes based on side effects to group patients and extract personalized rules. All three grouping schemes (side effects grouping first, action rules grouping first, meta-actions grouping first) led to different results. The best grouping was based on meta-actions first, and the resulting outcome was expected since meta-action are the primary connection between the action rules and the negative side effects.

In this work, we analyzed the dataset to mine meta-actions, and presented few ways of reducing negative side effects and extracting personalized action rules. Later, we will improve our work by using a larger dataset and introduce new evaluation metrics to analyze side-effects and improve patient care and treatments.

3.3 HCUP Dataset Description

In this chapter, we used the Florida State Inpatient Databases (SID) that is part of the Healthcare Cost and Utilization Project (HCUP). The Florida SID dataset contains records from several hospitals in the Florida State. It contains over 2.5 million visit discharges from over 1.5 million patients. The dataset is composed of five tables, namely: AHAL, CHGH, GRPS, SEVERITY, and CORE. The main table used in this chapter is the $Core$ table. The $Core$ table contains over 280 attributes; however, many of those attributes are repeated with different codification schemes. In the following experiments, we used the Clinical Classifications Software (CCS) that consists of over 260 diagnosis categories, and 231 procedure categories. This system is based on ICD-9-CM codes. In our experiments, we used fewer attributes that are described in this section. Each record in the $Core$ table represents a visit discharge. A patient may have several visits in the table. One of the most important attributes of this table is the $VisitLink$ attribute, which describes the patient's ID. Another important attribute is the Key, which is the primary key of the table that identifies unique visits for the patients and links to the other tables. As mentioned earlier, a $VisitLink$ might map to multiple Key in the database. This table reports up to 31 diagnoses per discharge as it has 31 diagnosis columns. However, patients' diagnoses are stored in a random order in this table. For example, if a particular patient visits the hospital twice with heart failure, the first visit discharge may report a heart failure diagnosis at diagnosis column number 10, and the second visit discharge may report a heart failure diagnosis at diagnosis column number 22. Furthermore, it is

Table 2: Mapping between attributes and concepts features.

Attributes	**Concepts**
VisitLink	Patient Identifier
DaysToEvent	Temporal visit ordering
DXCCSn	n^{th} Diagnosis, flexible attribute
PRCCSn	n^{th} Procedure, meta-action
Race, Age Range, Sex,..	Stable attributes
DIED	Decision Atribute

worth mentioning that it is often the case that patients examination returns less than 31 diagnoses. The *Core* table also contains 31 columns describing up to 31 procedures that the patient went through. Even though a patient might have gone through several procedure in a given visit, the primary procedure that occurred at the visit discharge is assumed to be the first procedure column. The *Core* table also contains an attribute called $DaysToEvent$, which describes the number of days that passed between the admission to the hospital and the procedure day. This field is anonymized in order to hide the patients' identity. Furthermore, the *Core* table also contains a feature called *DIED*, that informs us on whether the patient died or survived in the hospital for a particular discharge. There are several demographic data that are reported in this table as well, such as: Race, Age Range, Sex, living area, . . . etc. Table 2 maps the attributes from the *Core* table to the concepts and notations used in this chapter.

3.4 Completed Work Using HCUP Dataset

We used the HCUP surgical dataset SID to mine meta-actions based on action sets and develop a set of evaluation metrics. We also showcased some of the possible applications of such meta-actions and evaluated them using the HCUP dataset.

- Mining meta-actions based on action sets: In this work, we present a method to extract personalized meta-actions from surgical datasets with variable number of diagnoses or multivalued diagnoses. We also presented a meta-action representation with an ontology for action sets and defined their evaluation metrics. We used the Florida State Inpatient Databases (SID) that is a part of the Healthcare Cost and Utilization Project (HCUP) to demonstrate how to extract meta-actions and evaluate them.

- Side effects analysis and prediction: In this section, we give a brief description of the possible effects that might be considered as side effects or negative side effects. Moreover, we demonstrate how we can mine negative side effects based on action sets, and described how they can be evaluated. Furthermore, we show how patients negative action sets are used to cluster patients with similar negative side effects, and propose an incremental study to find the cluster membership of new patients.

- Meta-actions as a tool to evaluate action rules: In this work, we present the benefits of meta-actions in evaluating action rules in terms of two measures, namely Likelihood and Execution Confidence. In fact, in meta-actions, we extract real features values transition patterns, rather then a composing two feature values patterns. We also present an evaluation model of the application of meta-actions based on Cost and Satisfaction. We extracted action rules and meta-actions and evaluated them on the Florida State Inpatient Databases (SID) that is part of (HCUP) to evaluate our methodology.

In the following, we will start by presenting the core of our work which is meta-actions representation and mining. We will then further expend our research to side-effects analysis and give few examples of meta-actions applications in healthcare.

CHAPTER 4: META ACTIONS

Action rules can be seen as a tool for the analysis of transition patterns that inform decision makers about the possible changes to perform in order to reach a desired outcome. However, in order to move objects (patients) from their current population state to a more desirable population state, decision makers still need to acquire additional knowledge on how to perform the necessary changes, and what the triggers are for these changes. For instance, moving a patient from the sick population state to the healthy population state requires the practitioner to use a treatment such as an open heart surgery. This actionable knowledge is represented by meta-actions, and extracted from objects' state changes that actually occurred in the system.

In this chapter, we describe meta-actions with two different actionable data structures: action terms, and action sets. We demonstrate how meta-actions are mined and evaluated for the action terms representation, and show their limitations when multi-valued attributes are used. Furthermore, we demonstrate how to fix these limitation by mining the effects of meta-actions using action sets. We further evaluate the action sets extracted. As far as we know, this is the first attempt to mine meta-actions.

4.1 Meta-actions Overview

To build the strategies on top of the actionable tasks that action rules provide, we use meta-actions that are defined as follow:

*Definition 9 (Meta-actions) *Meta-actions associated with an information system S are defined as higher level concepts used to model certain generalizations of actions rules [37]. Meta-actions, when executed, trigger changes in values of some flexible features in S.*

More formally, let us define $\mathbf{M}(S)$ as a set of meta-actions associated with an information system S. Let $f \in F$, $x \in X$, and $M \subset \mathbf{M}(S)$, then, applying the meta-actions in the set M on an object x will result in $M(f(x)) = f(y)$, where object x is converted to object y by applying all meta-actions in M to x. Similarly, $M(F(x)) = F(y)$, where $F(y) = \{f(y) : f \in F\}$ for $y \in X$, and object x is converted to object y by applying all meta-actions in M to x for all $f \in F$.

Example 1. For example, let us take market segmentation for automobiles as a domain. Then, an automotive company, say company X, would divide its customers into: sedan cars seekers, sport cars seekers, wagon car seekers, all roads car seekers, and hash-backs cars seekers. An extensive list of classification features can be used to classify them. For instance, age would be a good feature that would inform us that a young person would prefer a hash-back car, and an older person is more likely to purchase sedan. Another feature such as number of kids would inform us that bigger families would prefer wagons or all roads rather than smaller cars, marital status could be a good indicator of sport cars preference, and so on. Other features can be analyzed for the purpose of customer satisfaction and segmentation. However, all features cited earlier are stable features and would not allow values transitions regardless of meta-actions applied. Let us assume another company Y, a luxurious car company, would like to acquire new customers, then their market segmentation would differ from X's market segmentation. In fact, the new segmentation would be based on new classification features, some of them are: customers income, car price range, customer functional needs, car comfort, car quality, and customers favorite brand. Given those features, the company can classify the customers based on their favorite brand, and would like to attract customers from other less luxurious brands. Based on those features, Y can apply a meta-action M that transitions car price range to match what customers can afford based on their income. This meta-action M can also affect car comfort and quality negatively to reduce production price while meeting customers' functional needs. In this example, Y's segmentation is based on luxurious cars, functional cars, powerful sport cars, and so on. A new customer segment, called entry luxury cars, can join the brand with the aid of meta-action M transitioning them from functional cars to entry-luxury cars.

Example 2. To give a new example, let us assume that classification features in S describe teaching evaluations at some school and the decision feature represents their overall score. Explaining difficult concepts effectively, Speaking fluent English, Stimulating student interest in the course, and Providing sufficient feedback are examples of classification features scored in the system. Then, examples of meta-actions associated with S will be: Changing the content of the course, Changing the textbook of the course, Posting all material on the Web. Clearly, those meta-actions will trigger changes in some of the features described such as providing sufficient feed back and stimulating students' interest in the course; however, none of these three meta-actions will influence the feature Speaking fluent English values will remain unchanged [37].

Example 3. Another example would be using Hepatitis as the application domain. Then, the increase in blood cell plagues and the decrease in level of alkaline phosphatase are examples of atomic action terms. Drugs like Hepatil or Hepargen are seen as meta-actions triggering changes described by these two atomic action terms [26, 38]. It should be noted that Hepatil is also used to get rid of obstruction, eructation, and bleeding. However, Hepargen is not used to get rid of obstruction, it is rather used to get rid of eructation and bleeding. Some of the effects of those meta-actions are not necessary and can be seen as side effects. At the same time, some needed changes are not triggered by the meta actions used and require the use of additional meta-actions.

Also, it should be mentioned here that expert knowledge concerning meta-actions involves only classification features. Now, if some of these features are correlated with the decision feature, then the change of their values will cascade to the decision through the correlation. The goal of action rule discovery is to possibly identify all such correlations.

Depending on the data organization and structure, meta-action effects can be mined and represented in two different ways. The first representation of meta-action effects is by actions terms when a fixed number of single-valued classification features are used as attributes. The second representation is by action sets that will be defined later in this chapter. The second representation is used when a variable number of multivalued-features are present.

4.2 Meta-actions Based on Action Terms

Meta-actions are actions, outside of the features F, performed by decision makers to transition objects from an initial known state with specific preconditions to a different state with known postconditions. The changes in flexible features, triggered by meta-actions in traditionally structured information systems, are represented by atomic action terms for the respective features, and reported by the influence matrix presented in [37].

4.2.1 Meta-actions Representation by Influence Matrices

Consider several meta-actions, denoted $M_1, M_2, \cdots, M_n$. Each one can invoke changes within values of some classification features in $F = \{f_1, f_2, \cdots, f_m\}$. The expected changes of values of classification features on objects from S triggered by these meta-actions are described by the influence matrix [36, 22] of the form $\{E_{ij} : 1 \leq i \leq n \;\&\; 1 \leq j \leq m\}$. Table 3 describes an example of an influence matrix associated with 6 meta-actions and three features: a, b, and c.

For instance, let us take meta-action M_2. It states that by executing M_2 on objects in S, two atomic action terms will be triggered. They are: $(a, a_2 \rightarrow a_1)$ and $(b, b_2 \rightarrow b_2)$. It means that objects in S satisfying the description $(a, a_2) \wedge (b, b_2)$ are expected to change their description to $(a, a_1) \wedge (b, b_2)$.

Table 3: Meta-actions influence matrix.

	a	b	c
M_1	-	b_1	$c_2 \rightarrow c_1$
M_2	$a_2 \rightarrow a_1$	b_2	-
M_3	$a_1 \rightarrow a_2$	-	$c_2 \rightarrow c_1$
M_4	-	b_1	$c_1 \rightarrow c_2$
M_5	-	-	$c_1 \rightarrow c_2$
M_6	$a_1 \rightarrow a_2$	-	$c_1 \rightarrow c_2$

Meta-actions are often available as domain knowledge and are documented as meta-data. However, they might be missing in case where domain experts are not aware of processes to solve unknown or new problems, or even undocumented in case where experts were not accustomed with meta-actions documentation methodology. Therefore, it is important to study different ways to mine meta-actions as well as extracting action rules. In fact, both concepts are interrelated and cannot be explored independently.

4.2.2 Meta-actions Mining

Mining meta-actions transformations requires the study of the transactional datasets in hand. Commonly, transactional datasets do not represent the objects temporal transformations resulting from applying meta-actions. Figure 1 summarizes the methodology followed in this thesis to mine meta-actions. To be able to mine meta-action's transformations (atomic action terms), objects have to be uniquely identified along with their transactions and clustered by their identifier (patient's ID). Object's transactions should be ordered based on temporal sequential order. Every two sequential object's transactions will be paired for every meta-action based on a temporal

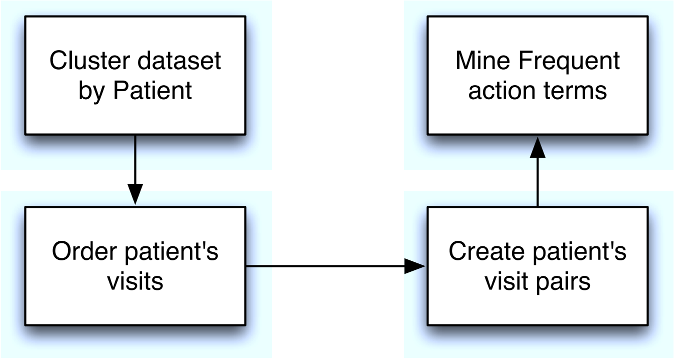

Cluster dataset by Patient
Order patient's visits
Create patient's visit pairs
Mine Frequent action terms

Figure 1: Meta-actions mining methodology.

precedence relationship. The resulting pairwise partitioning will model the atomic action terms transitions for each object given the meta-action applied. For instance, given a patient's visit recorded in our dataset with high blood pressure, high fever, and headache for his/her first visit to the doctor, who gave him/her a treatment m; at the second visit, the patient is diagnosed with high blood pressure, but shows no fever, and no headache. In such a case we can extract the following atomic action terms for this patient's pair of visits: $(fever, high \rightarrow no)$, $(blood\ pressure, high \rightarrow high)$, and $(headache, yes \rightarrow no)$.

Now, we introduce the set of transactions T. Let us assume that $S = (X, F \cup \{d\}, V)$ is a decision system, where X is a set of objects, F is a set of classification features, d is the decision feature, and V is a set of values for features in F, such that $f(X) \subseteq V$, for any $f \in F$. Also, let us assume that $\mathbf{M}(S)$ is a set of meta-actions associated with S. In addition, we define the set $\{s_{i,j} : j \in J_i\}$ of ordered transactions associated with $x_i \in X$, such that $s_{i,j} = [(x_i, F(x_i)_j)]$, where $(\forall i, j)[s_{i,j} \in T]$. The set $F(x_i)_j$ is defined as the set of feature values $\{f(x_i) : f \in F\}$ of the object x_i in the transaction uniquely represented by the transaction identifier j. Each transaction represents the current state of the object when recorded with respect to a temporal order based on j for all $s_{i,j} \in T$.

We define a precedence relationship denoted as " $>_p$ " on the system S to help locate the position of each transaction within each object's ordered transaction set. Given two transactions $s_{i,j}$ and $s_{i,k}$ for an object $x_i \in X$, the precedence relationship $s_{i,j} >_p s_{i,k}$ represents the order of the recorded transactions for the object x_i, and states that the transaction $s_{i,j}$ was recorded before the transaction $s_{i,k}$.

To strengthen this relationship, we define the set $P(S)$ of pairs $(s_{i,j}, s_{i,k})$ such that $(s_{i,j}, s_{i,k}) \in P(S)$ if and only if $s_{i,k}$ occurred directly after $s_{i,j}$ (there is no other transaction between them in the system S).

It should be observed that any pair $(s_{i,j}, s_{i,k}) = ([(x_i, F(x_i)_j)], [(x_i, F(x_i)_k)])$ in $P(S)$ represents a set of atomic action terms $\{(f, f(x_i)_j \rightarrow f(x_i)_k) : f \in F\}$.

We assume that there is always a set of meta-actions in M applied before any transaction $s_{i,k}$ with the exception of the very first transaction $s_{i,1}$ for each object $x_i \in X$. This suggests that the transaction $s_{i,k}$ in the pair $(s_{i,j}, s_{i,k})$ is a direct consequence of applying some set of meta-actions (let us say $m \in M$) to the object $x_i \in X$, and supports the assumption that the state of the objet x_i is being affected by these meta-actions.

As we already observed, each transaction pair $(s_{i,j}, s_{i,k})$ encompasses the set of atomic action terms $A_{j,k}$ of the form $\{(f, f(x_i)_j \rightarrow f(x_i)_k) : f \in F\}$, that defines a change of value $f(x_i)_j$ derived from the transaction $s_{i,j}$ to $f(x_i)_k$ derived from the transaction $s_{i,k}$, for each feature $f \in F$ and $x_i \in X$.

Now, we can use the sets $A_{j,k}$ to build the influence matrix covering all $m \in M$. Sets of action terms representing meta-actions in M can be built from sets of pairs in $P(S)$. Depending on the objects' states, some of the atomic action terms in $A_{j,k}$ may not be triggered by meta-actions. To be more precise, for a given meta-action m_j, only objects $x_l \in X$ that satisfy the following condition will be affected:

$$\exists (s_{i,j}, s_{i,k}) \in P(S) \text{ such that } F(x_l) \cap F(x_i)_j \neq \emptyset, \text{ where } s_{i,j} = (x_i, F(x_i)_j).$$

This way, by applying the meta-action m on x_i we will cover the set of feature values $\{F(x_l) \cap F(x_i)_j\}$, thus the underlying subset of atomic action terms will be triggered. This subset can be seen as an action term t containing a set of atomic actions with the domain $Dom(t) = \{a \in F : a(x_l) = a(x_i)_j\}$. Multiple action terms can be formed this way, however, not all possible action terms are applicable to a given dataset. The number of possible action terms for each meta-action grows monotonously with the number of extracted pairs, the features' domains sizes, and the number of transactions for each object in X.

Ultimately, in the worst case scenario every object is different and reacts differently to each specific meta-action. This might result in a large number of action terms for each meta-action. Possible conflicts within the same meta-action scope such as $(a, a_j \rightarrow a_k)$ and $(a, a_j \rightarrow a_l)$, or non-useful action terms such as $(a, a_j \rightarrow a_j)$ for $a \in F$ and $a_j, a_k, a_l \in V_a$ might be extracted. Not all atomic action terms are useful for all objects; however, it is important to keep a record of the different transitions for the sake of object personalized meta-actions.

We can evaluate different action terms, composing each given meta-action, to avoid conflicts and to use the more appealing ones to the examined object. Similar to the frequent itemsets used in the Apriori [1] algorithm, frequent action terms can be extracted from multiple pairs. Multiple action terms of different sizes can be formed from the resulting atomic action terms (pairs). Frequent action terms are characterized by their frequency of occurrence throughout all the meta-action partition of the dataset (all the meta-action pairs).

4.2.3 Meta-actions and Action Terms Evaluation

To evaluate the meta-actions, we need to evaluate the action terms composing them. A simple evaluation metric consists of the frequency of occurrence (or support) for each action term. Pairs, extracted from the data, share common atomic action terms transitions; thus, they share common action terms. For each action term t_j, we define its likelihood support $Like(t_j)$ as:

$$Like(t_j) = card(\{(s_{i,k}, s_{i,l}) \in P(S) : Left(t_j) \subseteq F(x_i)_k \text{ } and \text{ } Right(t_j) \subseteq F(x_i)_l\}) \tag{1}$$

where $(s_{i,k}, s_{i,l}) = ([(x_i, F(x_i)_k)], [(x_i, F(x_i)_l)])$, $x_i \in X$, $Left(t_j)$ is the left hand side of the frequent action term t_j, and $Right(t_j)$ is the right hand side of t_j.

The likelihood support of an action term measures the transition likelihood of their feature values but it neither takes into consideration the conflicts of action terms nor handles the meta-actions comparison in a normalized way. A different way to evaluate action terms is by computing their likelihood confidence, and thus, a possible meta-action confidence metric. The likelihood confidence of an action term t_j is computed as follow:

$$TermConf(t_j) = Like(t_j)/sup(Left(t_j)) \tag{2}$$

Given the set of atomic action terms $\{t_i : 1 \leq i \leq n\}$ composing a meta-action m_j, we can define the confidence of m_j as the weighted sum of its atomic action terms likelihood confidence, where the weights represent atomic action terms likelihood support. To be more precise, the meta-action confidence $MetaConf(m_j)$ is computed as follow:

$$MetaConf(m_j) = \frac{\sum_{i=1}^{n} Like(t_j) \cdot Conf(t_j)}{\sum_{i=1}^{n} Like(t_j)} \tag{3}$$

where n is the number of atomic action terms in m_j.

Note that some action terms will have the likelihood support below the required threshold value; therefore, they will not be considered as frequent action terms. However, those action terms are considered as outliers and it is important to keep track of them in the meta-actions for objects personalized meta-actions.

4.2.4 Tinnitus Meta-actions Mining

To mine the tinnitus meta-actions, we first generated the pairs of transactions for each patient's couple (pair) of sequential visits, we then clustered the dataset by meta-action based on the visits. Some of the patients' visits may be duplicated since the second visit of a patient is a consequence of applying a meta-action and might be as well an initial state of applying another meta-action. We used R 2.15 package software to generate the pairs and cluster the dataset. We obtained 36 pairs

Table 4: Tinnitus meta-actions evaluation.

	$Size1$			$Size2$		
$Action\ terms \rightarrow Meta-actions \downarrow$	# Action Terms	Avg Like	Avg Conf %	# Action Terms	Avg Like	Avg Conf %
SG	186	5	2	6360	2	36
HA	65	1	1	844	1	87
RC	225	52	17	18234	8	15

for the Sound Generator SG meta-action (treatment), only 3 for hearing Aid HA, no Combination CO of HA and SG, and as expected the majority 471 pairs for the Consultation RC.

The next part of this experiment consists of extracting the atomic action terms and generating the action terms of size larger than one. We use Weka 3.6.8 to extract the frequent action terms using the Apriori algorithm and consider each atomic action term as a feature (pair of features). We also implemented the algorithms described in this section and used them in our experiments. The results are summarized in Table 4 for action terms of size 1 and size 2.

It is absolutely possible to record all the results up to action terms of size 25 which is the number of features in our dataset; however, for the sake of space and discussion simplicity, we recorded only the action terms up to size 2. As can be noted from Table 4 the average likelihood grows monotonously with the number of pairs extracted from the dataset. Table 4 also shows that the average number of action terms of the meta action RC is the highest. This is justified by the high support that the meta-action RC displays. Table 4 shows a poor average confidence, and this is due to the high number of non-useful action terms. We can also note that the average confidence grows when the dataset size increases.

Meta-actions are the main core of action rules, whose execution depend on the quality and performance of meta-actions. It is then important to study meta-actions carefully in relation with the action rules discovery process. In this section, we proposed a meta-action mining mechanism to discover all their action terms for single-valued features information systems with fixed number of features. Furthermore, we introduced meta-actions main evaluation criteria. We implemented our proposed approach on the tinnitus handicap data, and evaluated the different meta-actions mined. We evaluated our approach limiting the size of the action terms mined for the meta-actions to two. Our experiments depict the meta-actions confidence and support.

4.3 Meta-actions Based on Action Sets

The changes in flexible features, triggered by meta-actions, are commonly represented by action terms for the respective features, and reported by an influence matrix presented in [37]. However, when an information system contains multivalued features where the same feature takes a set of values at any given object state and transitions to another set of values in a different object state, it is best to represent the transitions between the feature initial set of values and another set of values by action sets [34] that are defined as:

*Definition 10 (Action Set) *An action set in an information system S is an expression that defines a change of state for a distinct feature that takes several values (multivalued feature) at any object state.*

For example, $\{f_1, f_2, f_3\} \rightarrow \{f_1, f_4\}$ is an action set that defines a change of values for feature $f \in F$ from the set $\{f_1, f_2, f_3\}$ to the set $\{f_1, f_4\}$ where $\{f_1, f_2, f_3, f_4\} \subseteq V_f$. Action sets are used to model meta-action effects for information systems with multivalued features. In addition, the usefulness of action sets is best captured by the set intersection, between the two states involved, that models neutral action sets, and set difference, between the two states involved, that models positive action sets. In the previous example, neutral and positive action sets are respectively computed as follow: $\{f_1, f_2, f_3\} \rightarrow [\{f_1, f_2, f_3\} \cap \{f_1, f_4\}]$ and $\{f_1, f_2, f_3\} \rightarrow [\{f_1, f_2, f_3\} \setminus \{f_1, f_4\}]$.

We are studying surgical meta-action effects that trigger a change in the patients' state. The patients are in an initial state where the meta-actions are applied and move to a new posterior state. We use the set difference (positive action set) between two patient's states to observe the diagnoses that disappeared as a positive effect of applying meta-actions in the initial state. Furthermore, we use set intersection (neutral action set) to observe the diagnoses that remained the same; in other words, meta-actions applied had a neutral effect on these diagnoses.

4.3.1 Meta-actions Representation by an Ontology

This type of information concerning meta-actions is represented by an ontology (personalized) [6]. For instance, the example shown in Figure 2 models a meta-action composed of positive action sets which are labeled *positive* and neutral action sets which are labeled *neutral*. In addition, *positive* and *neutral* are composed of action sets respectively labeled $\underline{As_n}$ and $\overline{As_n}$, which in turn are composed of diagnoses labeled Dx_n.

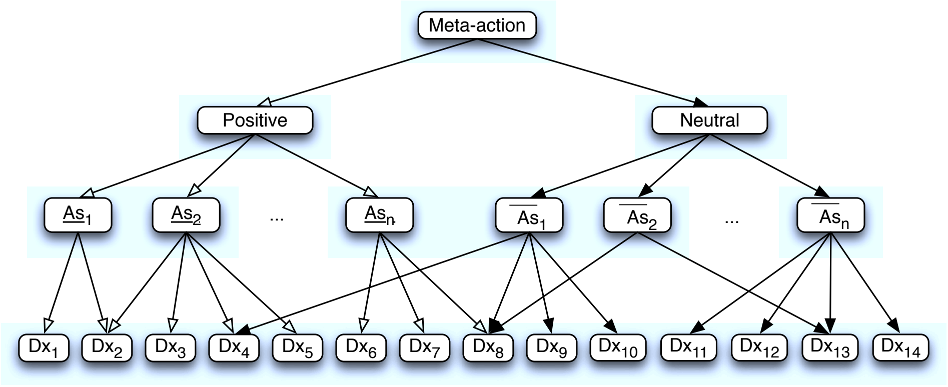


Figure 2: Ontology representation of a meta-action.

4.3.2 Meta-action Extraction

The effects of meta-actions, in the context of healthcare, represent the patient's state transition from an initial state to a different state. Those effects are mined from large datasets for each patient separately then merged together based on their common subsets to form state transition patterns. In other words, patients have to be uniquely identified along with their state transactions and clustered by their identifier. In this work, each state transaction for a patient represents a doctor consultation (patient visit to the doctor).

For each patient cluster, each transaction should be ordered based on its temporal sequential order. Every two consecutive patient's transactions will be paired for every meta-action based on a temporal precedence relationship. The resulting pairwise partition will model the effects of the applied meta-actions.

Given real life data representation in an information system, we defined two methods to extract meta-actions effects. The first method was defined in [33] and is used to extract meta-actions effects from traditional informations systems. In this method, since each feature has a different meaning and a single value at any given object state in the information system, meta-actions effects are represented with action terms and saved in an influence matrix to be used by practitioners. However, when multivalued features are used this method is useless; for instance, we cannot transition the diagnoses heart failure to the diagnoses skin problem. The second method is defined in this section and is used to extract meta-actions effects from information systems with variable number of features and multivalued features. This method is best suited for the surgical meta-actions mining problem since patients are diagnosed with several diagnoses at any consultation and have a different number of diagnoses.

Let us assume that $\mathbf{M}(S)$, where $S = (X, F, V)$, is a set of meta-actions associated with an information system S. In addition, we define the set $T = \{v_{i,j} : j \in J_i,\ x_i \in X\}$ of ordered transactions, patient visits, such that $v_{i,j} = [(x_i, F(x_i)_j)]$. The set $F(x_i)_j$ is defined as the set of feature values $\{f(x_i) : f \in F\}$ of the object x_i for the visit uniquely represented by the visit identifier j. Each visit represents the current state of the object (patient) when recorded with respect to a temporal order based on j for all $v_{i,j} \in T$. For any particular visit, the patient state is characterized by a set of diagnoses. Each diagnosis is seen as a feature, and each visit may have a different number of diagnoses.

For each patient's two consecutive visits $(v_{i,j}, v_{i,j+1})$, where meta-actions were applied at visit j, we can extract an action set. Let us define the set $P(S)$ of patient's two consecutive visits as $P(S) = \{(v_{i,j}, v_{i,j+1}) : \ x_i \in X, j \in J_i\}$. The corresponding action sets are: $\{(F(x_i)_j \to F(x_i)_{j+1}) : \ x_i \in X, j \in J_i\}$. We also define neutral action sets noted as $\overline{AS}$, and positive action sets noted as $\underline{AS}$. These action sets are: $\{(F(x_i)_j \to (F(x_i)_j \cap F(x_i)_{j+1})) : \ x_i \in X, j \in J_i\}$ and $\{(F(x_i)_j \to (F(x_i)_j \setminus F(x_i)_{j+1})) : \ x_i \in X, j \in J_i\}$ correspondingly, where $F(x_i)_j$ represents the set of diagnoses for a patient x_i at visit j.

The action sets resulting from applying meta-actions represent the actionable knowledge needed by practitioners. However, patients do not have the same preconditions and do not react similarly to the same meta-actions. In other words, some patients might be partially affected by the meta-actions and might have other side effects not intended by the practitioners. For this reason, we need to extract the historical patterns of action sets. Let us assume that neutral action set $\overline{as}_{i,j} = F(x_i)_j \cap F(x_i)_{j+1}$

and positive action set $\underline{as}_{i,j} = F(x_i)_j \setminus F(x_i)_{j+1}$, for any $x_i \in X$ and $j \in J_i$. Now, we define some properties for both the neutral and positive action sets extracted as follow:

1. $(\forall W)\ [W \subset \overline{as} \Rightarrow \overline{W} \in \overline{AS}]$
2. $(\forall W)\ [W \subset \underline{as} \Rightarrow \underline{W} \in \underline{AS}]$
3. $(\forall x_i \in X)\ (\forall j \in J_i)\ [\overline{as}_{i,j} \cup \underline{as}_{i,j} \subseteq F(x_i)_j]$
4. $(\forall x_i \in X)\ (\forall j \in J_i)\ [\overline{as}_{i,j} \cap \underline{as}_{i,j} = \emptyset]$

From the property number 1 and 2, given that any subset of an action set is an action set of the same meta-action, we can extract all action sets present in any pair of patient's visits using power sets. Let us define $\overline{P_{i,j}}$ as the power set of neutral action set $\overline{as_{i,j}}$ such that $\overline{P_{i,j}} \in \overline{AS}$. Similarly, we can define the set $\underline{P_{i,j}}$ as power set of positive action set $\underline{as_{i,j}}$ such that $\underline{P_{i,j}} \in \underline{AS}$. Hence, we can have all possible action sets composing a meta-action using power sets.

4.3.3 Meta-actions and Action Sets Evaluation

To evaluate the actions set patterns, we need to compute their frequency of occurrence for all patients. A good measure of frequency is the support and it is seen here as the likelihood of the occurrence for a specific action set (set of diagnoses disappearing or remaining). The likelihood $Like(\overline{as})$ of a neutral action set $\overline{as}$ is defined as follow:

$$Like(\overline{as}) = card(\{(v_{i,j}, v_{i,j+1}) \in P(S) :\ \overline{as} \in \overline{P_{i,j}}\}) \tag{4}$$

The likelihood $Like(\underline{as})$ of a positive action set $\underline{as}$ is defined as follow:

$$Like(\underline{as}) = card(\{(v_{i,j}, v_{i,j+1}) \in P(S) : \ \underline{as} \in \underline{P_{i,j}}\}) \tag{5}$$

The likelihood support of an action set measures the likelihood of features being affected by the meta-actions applied, but it does not provide a sense of how confident the action set is. A more sophisticated way to evaluate action sets would be to compute their likelihood confidence. The intuition behind the action set confidence lies in the normalization of the action set with regards to the patient's precondition. The likelihood confidence of a neutral action set $\overline{as}$ is computed as follow:

$$ActionConf(\overline{as}) = \frac{Like(\overline{as})}{card(\{v_{i,j} : \ \overline{as} \subseteq F(x_i)_j, \ \forall x_i \in X\})} \tag{6}$$

The likelihood confidence of a positive action set $\underline{as}$ is computed as follow:

$$ActionConf(\underline{as}) = \frac{Like(\underline{as})}{card(\{v_{i,j} : \ \underline{as} \subseteq F(x_i)_j, \ \forall x_i \in X\})} \tag{7}$$

Depending on the objects' states, some of the action sets in AS may not be triggered by meta-actions. To be more precise, for a given meta-action m, only objects $x_l \in X$ that satisfy the following condition will be affected:

$(\exists (v_{i,j}, v_{i,j+1}) \in P(S))(\exists v_{l,k} \in T)[F(x_l)_k \cap F(x_i)_j \neq \emptyset]$

Given the action sets composing a meta-action m, we can define the global confidence of m as the weighted sum of its action sets likelihood confidences where the weights represent action sets likelihood support. The intuition behind the meta-action confidence resides in defining how efficient the application of a meta-action is for any given patient's precondition. To be more precise, the meta-action confidence $MetaConf(m)$ is computed for both neutral and positive action sets as follow:

$$MetaConf(m) = \frac{\sum_{i=1}^{n} Like(as_i) \cdot ActionConf(as_i)}{\sum_{i=1}^{n} Like(as_i)} \tag{8}$$

where n is the number of action sets in m.

4.3.4 Surgical Meta-actions Mining for HCUP

We used our technique to extract meta-actions effects on the Florida SID dataset for several meta-actions. In this section, we report the confidence $ActionConf(as)$ and likelihood $Like(as)$ of few action sets for four different meta-actions. You can note from Table 5 that the positive action sets $ActionConf$ is very high, which means that patients' diagnoses disappear after applying meta-actions. In other words, the surgeries applied are very successful in curing patients diagnoses. On the other hand, the neutral action sets $ActionConf$ are small, which confirms the assumption that patients react in a consistently different way to meta-actions with regards to features that remained unchanged.

Table 5: Action sets confidence and likelihood.

Meta-action	**Action set**	**Type**	**ActionConf**	**Likelihood**
34	$\{127, 106\}$	Positive	88.88%	32
	$\{108\}$	Neutral	16.57%	29
43	$\{59, 55, 106\}$	Positive	85.71%	18
	$\{106\}$	Neutral	11.76%	18
44	$\{62, 106, 55\}$	Positive	94%	16
	$\{257, 101\}$	Neutral	15.62%	10
45	$\{59, 55\}$	Positive	89%	33
	$\{58\}$	Neutral	14.97%	28

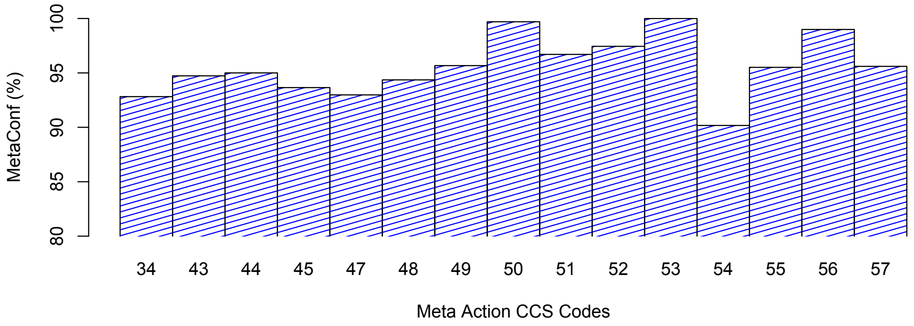


Figure 3: Meta-action confidence for surgical treatments.

Moreover, the likelihood of neutral action sets extracted is small, which means that very few diagnoses remain unchanged after the surgeries. Table 5 represents the meta-actions (procedures) and action sets elements (diagnoses) with their CCS codification [3].

In addition, we report in Figure 3 the meta-action confidence for 15 different meta-actions. We show in Figure 3 that the meta-actions are consistently successful for all their action sets regardless of the patients preconditions for these meta-actions. Figure 3 shows meta-actions with their CCS codes [3].

Mining surgical meta-actions is a hard task because patients may react differently to the applied meta-actions, and surgeries outcomes are different from one patient to another. In this section, we presented a meta-action effects mining technique for surgical datasets with variable number of diagnoses (multivalued features). Furthermore, we presented the ontology representation of meta-action effects, and used the SID dataset that is part of HCUP to demonstrate the usefulness of our methodology in comparison with the action terms based techniques.

4.4 Chapter Conclusion

In this chapter, we introduced the core model for this dissertation which is refining the structure of actionability and introducing the actions and their effects representation for meta-actions. We explored the action terms and action sets structures of meta-actions and their influence matrix and ontology representations. Furthermore, we explored the different ways to mine and extract meta-actions effects, and modeled their evaluation metrics. To demonstrate the usefulness and stability of our model, we extracted and evaluated meta-actions effects from two large datasets. In fact, we extracted meta-actions effects from the Tinnitus handicap dataset that is traditionally structured with single valued features and a fixed number of features, and from the HCUP Florida SID dataset that contains multivalued features. Our results show that we have a good average action term confidence and action set confidence as well as a good meta-action confidence for both representations.

CHAPTER 5: SIDE-EFFECTS ANALYSIS AND THEIR USABILITY

Meta-actions are actions that trigger certain changes in objects' states. These changes are commonly referred to as meta-action effects that affect certain properties of the examined objects. Meta-actions effects can be positive, neutral, or negative. Positive effects help objects positively to transition them into a more desired state. Neutral meaning does not introduce any effects on the overall state of the objects. Negative effects that may possibly harm or move the object into an undesired state. These effects can be seen as side effects when they are not intended by users.

The side effects are not only related to the applied meta-actions, but also to the objects' initial state. For this reason, it is important to study the objects in hand and anticipate the possible side effects when applying specific meta-actions. For instance, in health care, patients are mainly represented by their diagnoses; however, their demographic data such as age and gender or additional data such as co-morbid conditions may possibly affect the application of certain treatments (meta-actions). In this section, we focus on the evaluation and possible prediction of potential side effects given an object and the applicable meta-actions.

Meta-action effects are represented by two actionable data structures dividing the study of side effects into: action term based side effects, and action sets based side effects. Both representations of side effects are described and analyzed in this section.

5.1 Side Effects Based on Action Terms

As stated before, the main goal of meta-actions is to trigger action rules. However, it is often the case that when applying meta-actions for the purpose of executing a specific action rule, a set of additional unrelated and potentially harmful atomic action terms is triggered. The additional action terms resulting from the meta-action application are called side effects. Meta-actions might move the values of some object's features from negative to positive $f(x) \in E_n(x)$ and $f(y) \in E_p(y)$ (desirable positive side effects), and values of some object's features from positive to negative values $f(x) \in E_p(x)$ and $f(y) \in E_n(y)$ (undesirable negative side effects). Even though the features transitioning from positive to negative values might result in catastrophic situations, they were not fully investigated in previous work involving action rules discovery. In the following, we depict two types of side effects based on action terms and we give a brief description for each type.

5.1.1 Meta-actions Side Effects

Side-effects based on action terms in the context of meta-actions alone are the effects that occur for specific small clusters of objects. This type of side effects is discovered in the meta-action extraction process. It is represented by the action terms that exhibit very low or unusual likelihood of occurrence. In fact, this type of action term is very rare in our dataset, and it was extracted from a very small number of objects. We can think of this type of effects as minor effects of a meta-action that do not represent the core goal of applying this meta-action. Detecting this type of side effects is done by setting a minimum likelihood for the action terms, or setting a minimum jump in values of likelihood between the action terms.

5.1.2 Action Rules Side Effects

Side-effects based on action terms in the context of action rules are the unintended changes in the values of some flexible features that meta-actions trigger on objects. In other words, those effects are triggered by meta-actions but are outside of the intended action rule scope. To discover those side effects, we can perform two set operations. We start by performing a set difference operation between the antecedent side of the action rule and the meta-actions' action terms reported in the influence matrix. The result is then intersected with the object's precondition to get the final set of side effects.

5.2 Side Effects Based on Action Sets

In healthcare, the study of side effects is mainly related to treatments and patients' conditions. In this section, we study the representation of side effects with regards to action sets based meta-actions. Meta-actions mined from information systems with variable number of features and multi-valued features result in positive and neutral effects that were studied in Chapter 4. In addition, they might result in negative side effects when applied to specific patients with particular precondition states. For instance, applying meta-action treatment m to patient x who is diagnosed with $F(x)_t = \{Dx_1, Dx_2, Dx_3\}$ at the precondition state time t might transition the patient to a new state with the following diagnoses $F(x)_{t+1} = \{Dx_1, Dx_4\}$ at time $t+1$ and introduce a new diagnosis Dx_4 that was not present before applying m. In this example Dx_4 is seen as a negative side effect that appeared as a result of applying

m to x.

In the remainder of this section, we study the negative action sets (negative side effects) mining and representation. We also show how to cluster patients based on these negative action sets and analyze the clusters. Furthermore, we attempt to predict treatments' negative side effects for patients using an incremental study of the clustering.

5.2.1 Negative Action Sets Mining

Negative side effects are represented by action sets which model the appearance of certain diagnoses when applying a meta-action on specific patients. These diagnoses were not intended by the physician and can be harmful to the patient. The negative action sets are part of the meta-action effects described in Section 4.3, and they are best captured by the reverse set difference between the prior and posterior state of the patient. For example, let us apply a meta-action m to a patient x with the prior state $F(x)_t = \{Dx_1, Dx_2, Dx_3\}$, and assume a posterior postcondition state $F(x)_{t+1} = \{Dx_1, Dx_4\}$ as a result, then the action set resulting is described by: $\{Dx_1, Dx_2, Dx_3\} \rightarrow [\{Dx_1, Dx_4\} \setminus \{Dx_1, Dx_2, Dx_3\}]$, where $[\{Dx_1, Dx_4\} \setminus \{Dx_1, Dx_2, Dx_3\}]$ represents the reverse set difference between the left hand side of the action set and its right hand side. Negative side effects are represented by an ontology as described in Section 4 with the addition of negative action sets labeled $\overline{\underline{As}}$ an represented in red in Figure 4.

Once we define the format of negative action sets, we use the same action set mining technique described in Section 4.3.2. In fact, we order patients by visit date,

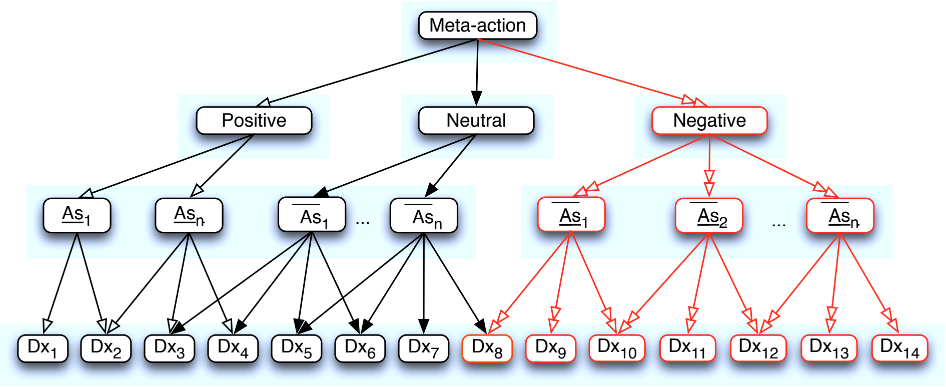


Figure 4: Ontology representation of a meta-action with negative effects.

and create pairs containing two consecutive visits for each patient. The negative action sets are then extracted from those pairs for each patient and a power set is then generated to extract all possible combinations.

5.2.2 Patients' Clustering Based on Side Effects

Patients react similarly to some treatments that result in the same negative side effects. Therefore, it is important to keep track of the meta-actions side effects and cluster the patients who experience similar negative side effects. Clustering the patients based on the negative side-effects is done using the negative action sets. The negative action sets are supported by patients that reacted negatively to the treatments (meta-actions) applied with the respective side effects. The supporting patients for a specific negative action set constitute a cluster of patients; hence, the clustering is done by grouping those patients.

By the supporting set for a negative action set $\overline{\underline{As}} = [F(x)_{t+1} \setminus F(x)_t]$ of the form $F(x)_t \rightarrow [F(x)_{t+1} \setminus F(x)_t]$ in an information system $S = (X, F, V)$, where $F(x)_t = \{f(x)_t : f \in F\}$, we mean the set of patients $x \in X$ represented by the expression $sup(\overline{\underline{As}}) = \{x \in X : (\forall f(x) \in \overline{\underline{As}})\ [(f(x) \in F(x)_{t+1}) \wedge (f(x) \notin F(x)_t)]\ \}$.

Now, $sup(\overline{\underline{As}})$ represents the set of the objects affected by the association. This way each supporting set of patients represents a different cluster $sup(\overline{\underline{As_i}})$ labeled by $\overline{\underline{As_i}}$.

5.2.3 Negative Side Effects Evaluation

The negative actions sets are evaluated and analyzed using the same metrics developed in Chapter 4 to evaluate neutral and positive action sets. In fact, for each negative action set $\overline{\underline{As}}$, we can compute the likelihood $Like(\overline{\underline{As}})$ as follows:

$$Like(\overline{\underline{As}}) = card(sup(\overline{\underline{As}})) \tag{9}$$

The likelihood $Like(\overline{\underline{As}})$ is a good measure of the spread or dominance of this negative side effect for a specific meta-action.

Of course, we can also compute the negative action set confidence $ActionConf(\overline{\underline{As}})$ as follows:

$$ActionConf(\overline{\underline{As}}) = \frac{Like(\overline{\underline{As}})}{card(\{x_i \in X : \; [\overline{\underline{As}} \subseteq F(x_i)_{t+1}]\})} \tag{10}$$

where $F(x_i)_t$ represents the initial state of the object or patient x_i. Note here that the $ActionConf(\overline{\underline{As}})$ does not model the confidence of predicting that patients with the initial state $F(x_i)_t$ will react with negative side effects to the meta-action; it rather models the confidence of the action set being a negative action set $\overline{\underline{As}}$ and not a neutral one. In other words, it does not model correlation between $F(x_i)_t$ and $\overline{\underline{As}}$.

In Chapter 4, we computed the meta-action confidence $MetaConf(m)$ for m to acquire a general idea on its stability with regards to patients initial states and the positive and neutral action sets. However, we did not include the negative side effects because the purpose of the meta-action is to cure the initial diagnoses of the patients.

On the other hand, the negative side effects may have originated from the correlation between the applied meta-action and some stable or unknown features and not necessarily from the initial state of the patient.

$$NegMetaConf(m) = \frac{\sum_{i=1}^{n} [Like(\underline{\overline{As_i}}) \cdot ActionConf(\underline{\overline{As_i}})]}{\sum_{i=1}^{n} Like(\underline{\overline{As_i}})} \tag{11}$$

where n is the number of extracted negative action sets. The negative meta-action confidence informs us about how the initial state of the patient is correlated with the negative action sets.

5.2.4 Negative Side Effects Predictions

An important concern in healthcare is about the possibility of predicting side effects of meta-actions for specific patients. In fact, we noticed that similar patients with similar initial states have different reactions to the same treatments. Some of the patients have severe negative side effects and some do not have any side-effects, which makes the problem even more challenging.

Once negative action sets are mined for a specific meta-action m, we can then analyze them and evaluate their correlation to the patient's state. However, we would like to predict the unknown negative side effects for the patients resulting from the application of m. To do so, we use the patients clusters based on the negative actions sets. We first group patients by their side effects using their negative action sets. The resulting clusters of patients labeled by negative side effects will then serve as models for future patients that have to undergo the meta-action m.

We build an incremental study for the clustering approach to include new patients in the clusters they belong to. Our assumption is that a patient who is about to undergo a meta-action m belongs to the cluster which contains the most similar patients.

Similarity is a very important factor in the prediction, and it is computed using the patient's stable and flexible features $F = F_{st} \cup F_{fl}$. To compute the similarity $w_{i,j}$ between two patients x_i and x_j, we build a feature vector for the patients' features based on their type. For single-valued features, we use a binary value such that $w_{i,j}(f) = 1$ for similar feature values and $w_{i,j}(f) = 0$ for different ones. For multi-valued features, we use a none symmetric similarity based on the proportion of similar feature values. For instance, if $F(x_i) = \{Dx_1, Dx_2, Dx_3, Dx_4, \}$ and $F(x_j) = \{Dx_1, Dx_2, Dx_5\}$ and we would like to predict the side-effects for patient x_i given $\underline{\overline{As(x_j)}}$, then the similarity for a multi-valued feature f is computed as follows:

$$w_{i,j}(f) = \frac{card(F(x_i) \cap F(x_j))}{card(F(x_i))} \tag{12}$$

Once we have computed the individual similarities of all features, we can compute the similarity $w_{i,j}$ between two patients' feature vectors using Manhattan similarity or RBF kernel described as follows:

$$w_{i,j} = e^{-(\|x_i - x_j\|^2/2\sigma^2)} \;\; \forall x_i \neq x_j \tag{13}$$

After computing the new patient's similarity with all members of a cluster c_k for the negative action set As_k, we can compute the mean of each cluster similarity to the new patient x_i. The new patient is then assigned to the cluster with the highest mean similarity.

Another model suggests the computation of the mean vector centroid of the cluster, used in K-means like clustering algorithms [11], once we have all its members, then compute the distance between the new patient x_i and each cluster mean vector centroid. The new patient x_i is then assigned to the cluster with the smallest distance.

5.2.5 Experimental Evaluation

We performed several experiments regarding mining negative action sets and analyzing patients' clusters based on negative side effects. We used four meta-actions to extract and analyze side effects. The four meta-actions used are referenced by the following procedure CCS codes [3]: 34, 43, 44, and 45.

We started by mining the negative action sets and analyzing them with the evaluation criteria described in Subsection 5.2.3. We also give a few examples of the negative action sets mined in Table 9. We then grouped the patients based on side effects and we analyzed the resulting clusters.

Table 6: Negative clusters analysis for all meta-actions.

Meta-action	Average clusters size	Total number of clusters	Average Likelihood	Average ActionConf	MetaConf
45	4.84	405931	1.69	0.84	0.53
44	4.46	178446	1.49	0.83	0.55
43	4.70	194957	1.37	0.85	0.62
34	4.63	180372	1.44	0.83	0.58

You can note from Table 6 that the $MetaConf$ is smaller than the average confidence since it reflects a better global confidence of the meta-actions with regard to negative action sets' *Likelihood*. However, Figure 5 displays the proportion of action sets confidence by meta-action, and shows that more than 73% of action sets have

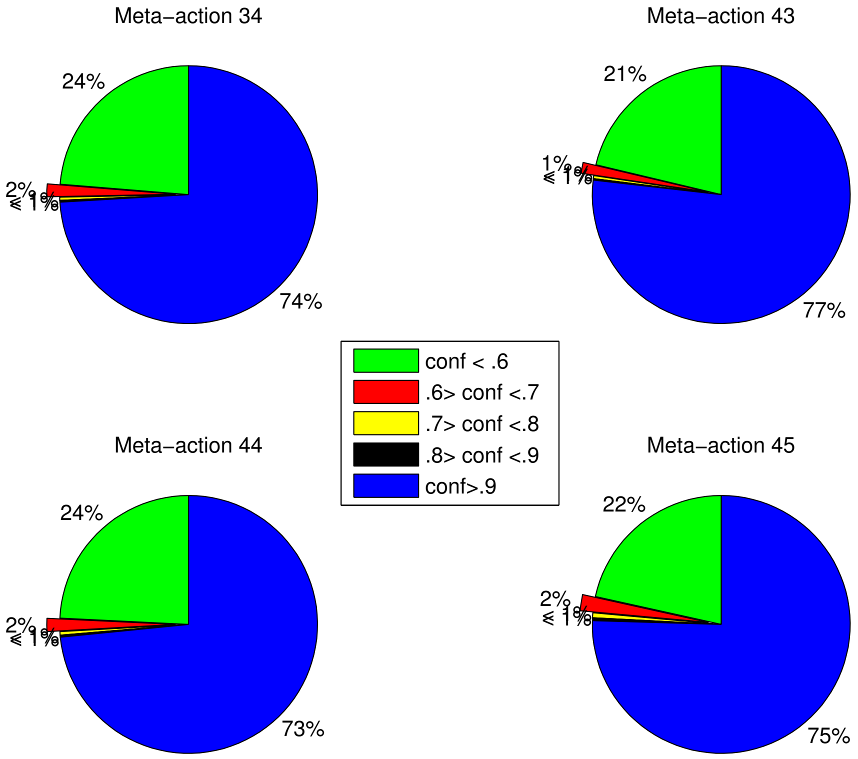


Figure 5: Negative action sets confidence proportion.

over 90% confidence for all meta-acions. The total number of clusters, the average likelihood, and the average cluster's negative action set size are also reported in Table 6 for each meta-action.

Tables 7, 8, 10, and 11 show that the number of clusters follows a Gaussian distribution [9] behavior with regard to their negative action sets sizes. In addition, the average likelihood decreases when the size of the cluster's action sets increases. The action set cluster with size 0 indicates no side effects; in other words, patients in this cluster did not have any side effects as a result of applying the meta-action. Those tables show that increasing the size of the cluster action sets, in most of the cases,

Table 7: Negative clusters analysis for meta-action 34.

Meta-action	Action set clusters size	Number of clusters	Average Likelihood	Average ActionConf
34	0	1	2264	1
	1	213	60.11	0.73
	2	5803	5.86	0.65
	3	29814	1.86	0.67
	4	52691	1.17	0.78
	5	47620	1.02	0.89
	6	28146	1.00	0.96
	7	11794	1.00	0.99
	8	3495	1.00	0.99
	9	703	1.00	1
	10	87	1.00	1
	11	5	1.00	1

increases the average of $ActionConf$. This is due to introducing more constraints in the action sets. Figure 6 summarizes the trend of the action sets average confidence in a better way. This figure shows that the average action set confidence is low for action sets' clusters with sizes ranging from 1 to 4. This is due to the small number of supporting patients for these clusters.

Table 8: Negative clusters analysis for meta-action 44.

Meta-action	Action set clusters size	Number of clusters	Average Likelihood	Average ActionConf
44	0	1	3905	1
	1	219	82.39	0.80
	2	6912	5.99	0.67
	3	36008	1.67	0.67
	4	54893	1.11	0.79
	5	43770	1.01	0.91
	6	23878	1.00	0.97
	7	9463	1.00	0.99
	8	2700	1.00	0.99
	9	532	1.00	0.99
	10	66	1.00	1
	11	4	1.00	1

Table 9: Examples of negative action sets for all meta-actions.

Meta-action	Negative action set	Size	Likelihood	ActionConf
34	[1]	1	404	0.99
	[238]	1	308	0.83
	[134]	1	719	0.88
	[155]	1	932	0.84
	[1 , 155]	2	187	0.89
	[134 , 155]	2	366	0.77
	[134 , 1 , 155]	3	54	0.87
	[134 , 59 , 155]	3	55	0.58
43	[257]	1	429	0.93
	[254]	1	399	1.00
	[155]	1	238	0.91
	[159]	1	227	0.81
	[259 , 254]	2	102	0.89
	[257 , 254]	2	72	0.96
	[113 , 159 , 254]	3	10	1.00
	[105 , 254 , 155]	3	8	1.00
44	[254]	1	403	1.00
	[102]	1	276	0.99
	[197]	1	272	0.93
	[3]	1	214	0.98
	[2 , 244]	2	111	0.86
	[197 , 238]	2	121	0.68
	[134 , 2 , 244 , 249 , 157]	5	3	1.00
	[2 , 52 , 249]	3	12	1.00
45	[102]	1	1532	0.97
	[130]	1	495	0.91
	[153]	1	456	0.91
	[60]	1	452	0.96
	[2 , 244]	2	252	0.90
	[153 , 60]	2	127	0.95
	[146 , 120]	2	68	0.96
	[2 , 244 , 249]	3	58	0.85

Table 10: Negative clusters analysis for meta-action 43.

Meta-action	Action set clusters size	Number of clusters	Average Likelihood	Average ActionConf
43	0	1	2645	1
	1	213	64.12	0.78
	2	6277	5.46	0.67
	3	33158	1.64	0.68
	4	54737	1.10	0.80
	5	48576	1.01	0.92
	6	30447	1.00	0.97
	7	14476	1.00	0.99
	8	5275	1.00	0.99
	9	1457	1.00	0.99
	10	296	1.00	1
	11	41	1.00	1
	12	3	1.00	1

Table 11: Negative clusters analysis for meta-action 45.

Meta-action	Action set clusters size	Number of clusters	Average Likelihood	Average ActionConf
45	0	1	12076.0000	1
	1	225	225.2400	0.77
	2	8230	13.2682	0.63
	3	59864	2.4103	0.62
	4	111169	1.2360	0.77
	5	104193	1.0493	0.91
	6	70782	1.0130	0.97
	7	35255	1.0034	0.99
	8	12466	1.0006	0.99
	9	3143	1.0000	0.99
	10	542	1.0000	1
	11	58	1.0000	1
	12	3	1.0000	1

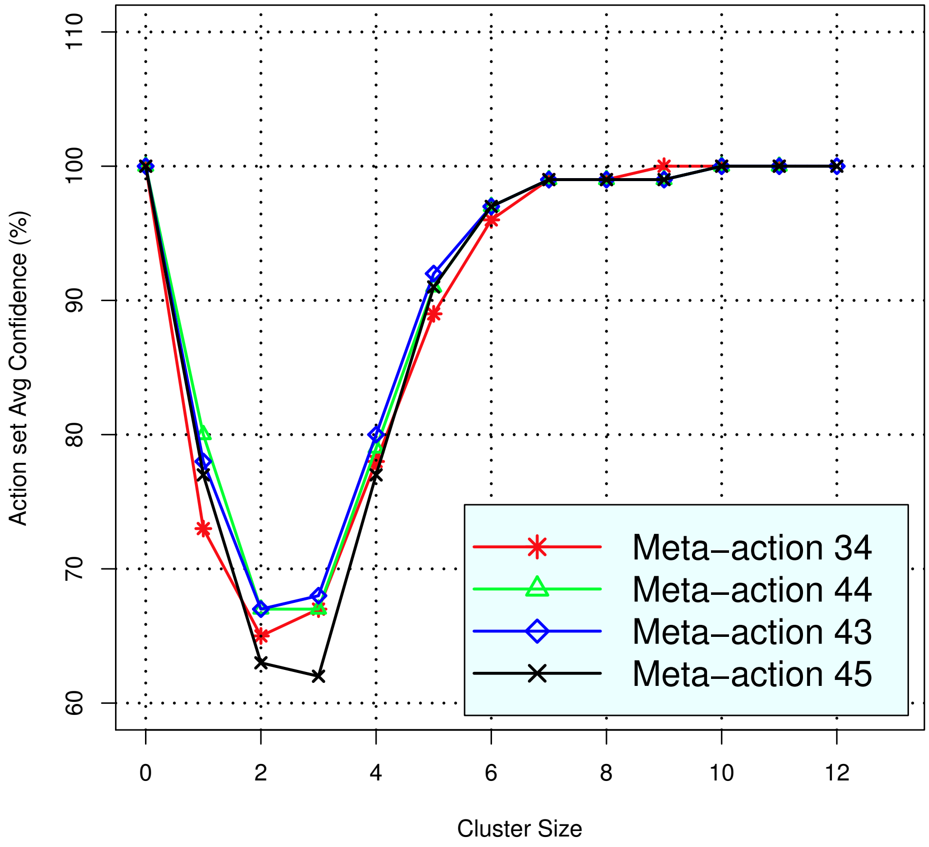


Figure 6: Negative action sets confidence by size of clusters.

Mining negative side-effects allows us to cluster patients with similar negative action sets in order to extract personalized action rules. The purpose of personalized action rules is to decrease the negative side effects. We have demonstrated in this section how negative side effects based on action terms are represented and how negative action sets are structured and extracted. We then presented negative action sets evaluations metrics, and analyzed patients' clusters based on these metrics. We also presented an incremental clustering scheme similar to K-means [11] using the similarity among patients and the clusters centroid.

5.3 Personalized Action Rules Extraction Based on Object Clustering

In this chapter, we study closely the side effects of applying meta-actions. We acknowledge that those negative side effects are not avoidable in most situations; therefore, we strive to personalize the action rules and their respective meta-actions applied to objects based on their reactions to meta-actions. We strongly believe that action rules should be extracted from data sets describing objects that have the same negative side effects, and we present three objects grouping techniques based on negative side effects. We believe that personalized action rules can be extracted based on the three dimensions: meta-actions, objects and objects' negative side effects. Figure 7 depicts object grouping based on meta-actions and side effects. In this chapter we used the tinnitus handicap dataset that was exploited in previous research [40] to extract action rules. By analyzing the patients negative side effects, and grouping patients based on their reactions to treatment (meta-actions), we can extract personalized action rules. We compred the three grouping schemes and deduced that the meta-action based grouping result in the best personalized action rules.

5.3.1 Personalized Object Grouping

Unfortunately, the negative side effects resulting from applying meta-actions are unavoidable in most situations. However, we can still lower the negative side effects resulting from executing the meta-actions by personalizing the action rules applied to objects. Action rules dictate the sets of meta-actions to apply to be triggered. There are multiple subsets of meta-actions that could be applied to different action rules and result in multiple subsets of negative side effects. We aim to minimize the

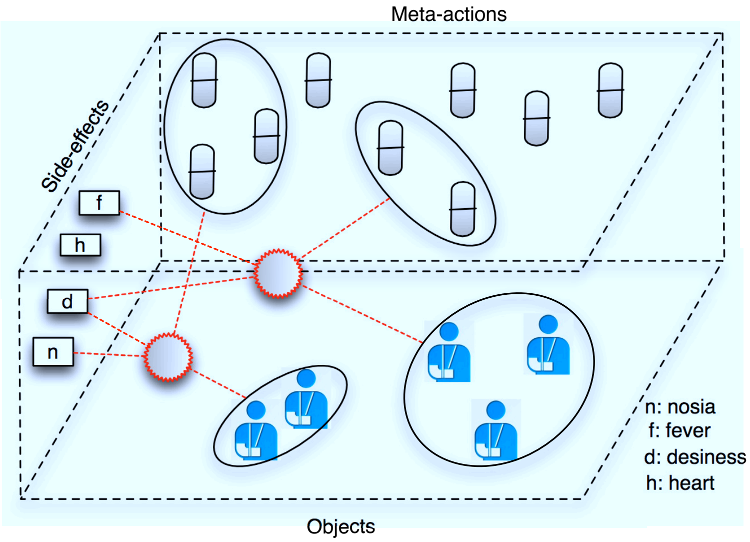


Figure 7: Three dimensional representation of object grouping.

negative side effects for a large number of objects by discovering personalized action rules while keeping their utility and increasing their support and confidence. In the following subsections, we define three techniques to group objects based on their side effects resulting from applying meta-actions for a personalized action rules discovery system.

5.3.1.1 Side Effects Based Grouping

In this technique, as the title suggests, we group objects based on side effects. In most situations, objects have known side effects such as patients having allergies. However, more side effects can be determined based on the possible meta-actions applied. Let us define the set of meta actions $M(S)$ on the system $S = (X, F \cup \{d\}, V)$, such that $F(x) = \{f(x) : f \in F\} = E_n(x) \cup E_p(x)$ for $x \in X$. We aim at grouping objects $x \in X$ that have the same negative side effects for any meta-action in $M = \{M_k\}_{k \in K} \subseteq M(S)$ where $K = \{1, \ldots, |M|\}$. This grouping will result in a partition of X defined by the equivalence relation given in the following:

$$x_i \approx_M x_j \; iff \; (\forall k \in K)[E_{n,k}(y_i) = E_{n,k}(y_j)] \text{ where}$$
$$[M_k(F(x_i)) = F(y_{i,k})], \; [M_k(F(x_j)) = F(y_{j,k})], \; and \; [x_i, x_j \in X].$$

Also, we assume here that x_i is converted to $y_{i,k}$ and x_j is converted to $y_{j,k}$ by $M_k \in M$. In a real setting, as explained earlier, we need to follow a set of steps in order to group objects in the most optimal way minimizing the negative side effects to extract the right action rules. In fact, meta-actions result in different negative side effects from one object to another. We defined the following steps for personalizing action rules based on common negative side effects:

- We first need to extract the negative side effects from applying the meta-actions for each object $x \in X$ if they are not defined yet. This process is performed by analyzing our decision system S and extracting all negative side effects $E_n(x)$ for all $x \in X$ for each transaction (or observation) that happens directly after applying the meta-actions.

- We then group the objects that have the same side effects using the previously defined equivalence relation that will result in a partition G. Those groups encompass the objects observations in our decision system for each object in the group.

- We extract the personalized action rules for each group $g \in G$ using the objects observation in each group and the algorithms presented in [24, 23].

- Finally, we select the personalized action rules with the best support and confidence pair to apply in each group.

The personalized action rules extracted for each group of objects will result in the same negative side effects when applying the related meta-actions to trigger the action rules. However, grouping the objects by negative side effects first will decrease the object population for discovering action rules for each group. This might result in decreasing the support of the rules or even failing to discover all possible rules. To remediate to this problem, we propose an action rule based grouping that is described in the following.

5.3.1.2 Action Rules Based Grouping

Another way of grouping objects in X is by grouping action rules with respect to their common support. Let us assume that AR_S is a set of action rules extracted from a decision system $S = (X, F \cup \{d\}, V)$, and $r_1 = r_1(r_{11}, r_{12})$ $r_2 = r_2(r_{21}, r_{22}) \in AR_S$. The binary relation $\equiv_S \subseteq AR_S \times AR_S$ is defined as follows:

$$r_1 \equiv r_2 \; iff \; [sup(r_{11}) = sup(r_{21})].$$

Clearly, $\equiv_S$ is an equivalence relation which partitions the action rules in AR_S into classes such that any two action rules in the same class have the same set of supporting objects in X. For instance, let us assume that $r_1 = r_1(r_{11}, r_{12}) \in AR_S$, $r_{11} = [t_{11} \rightarrow t_{12}]$, and $sup(r_{11}) = Y_1$. Also, we assume here that $r_2 = r_2(r_{21}, r_{22}) \in AR_S$, $r_{21} = [t_{21} \rightarrow t_{22}]$, $sup(r_{21}) = Y_2$, and $r_1 \equiv_S r_2$. Then, $Dom(t_{11}) = Dom(t_{21})$ and $F(Y_1)/Dom(t_{11}) = F(Y_2)/Dom(t_{21})$, where $F(Y_1)/Dom(t_{11})$ is the set of values of attributes listed in t_{11}. Now, if $t_{11} = (f_1, v_{11}) \wedge (f_2, v_{21}) \wedge \cdots \wedge (f_k, v_{k1})$, then $F(Y_1)/\{f_1, f_2, \ldots, f_k\} = \{(f_1, v_{11}), (f_2, v_{21}), \ldots, (f_k, v_{k1})\}$ displays all properties for which objects in X have to satisfy in order to be affected by r_1 or r_2.

So, the relation $\equiv_S \subseteq AR_S \times AR_S$ partitions the action rules in AR_S into equivalence classes in such a way that each class $[r_1]_{\equiv} S$, where $r_1 = r_1(r_{11}, r_{12})$ and $r_{11} = [t_{11} \rightarrow t_{12}]$, has a unique set of attribute values $I([r_1]_{\equiv_S})$ for $Dom(t_{11})$ which is used as its identifier. In the section above, $I([r_1]_{\equiv_S}) = \{(f_1, v_{11}), (f_2, v_{21}), \ldots, (f_k, v_{k1})\}$. Each identifier defines a subset of all objects in X satisfying all properties listed in it. This way, we do not only group the objects in X but also identify the largest subset of objects in X which can be affected by a minimum of one action rule. We still need to further partition the obtained groups of objects by taking into consideration personalized action rules based on their negative side effects. We use grouping mechanism similar to the side effects based grouping, presented in the previous section. We define the following steps for personalizing action rules based grouping with respect to the negative side effects:

- We first extract the set of action rules AR_S from S, next we group the objects in X based on the equivalence relation $\equiv_S \subseteq AR_S \times AR_S$. Lets assume that $G = \{G_k\}_{k \in K}$ represents that grouping.

- We extract the negative side effects $E_n(x)$ for all $x \in X$ resulting from applying the meta-actions to trigger the action rules in $AR_S(x)$ for each object $x \in X$.

- Then, we split each group $G_k \in G$ in such a way that objects having the same side effects with respect to action rules associated with G_k are placed in the same sub-groups. For that purpose, we use the equivalence relation $\approx_M$ defined in the previous section. It will result in new sub-groupings G_{ki} of X.

- We merge the sub-groups G_{ki} if their respective action rules trigger the same negative side effects.

- Finally, we select the personalized action rules with the best support and confidence pair to apply in each group.

This grouping will not generate smaller groups than the previous technique since the merging step insures that groups with the same negative side effects are merged together.

5.3.1.3 Meta-actions Based Grouping

The previous grouping technique based on action rules does not take the meta-actions into consideration. This may result in groups of objects with their respective action rules having different meta-actions applied to trigger their rules. Another grouping technique groups objects first with respect to meta-actions applied to them. Clearly, meta-actions are not applied randomly to objects; they are either applied based on an action rule needs, or applied by an administrator making a decision based on his/her expertise. Let us have a decision system $S = (X, F \cup \{d\}, V)$ and a set of meta-actions $M(S)$ associated with S which can be applied to objects in X. We first group objects by meta-actions and then we split the obtained groups further with respect to the negative side effects which are triggered by them. In order to group objects in X with respect to a meta-action $M_p = \{E_{pk} : k = 1, 2, \ldots, m\} \in M(S)$, where E_{pk} are atomic actions triggered by M_p, we assume that $Dom(M_p) = \{Dom(E_{pk}) : k = 1, 2, \ldots, m\}$ and define the following relation for any $x_i, x_j \in X$:

$$x_i \approx_{M_p} x_j \; iff \; [(F(x_i) = F(x_j)) \wedge M_p(F(x_i)) = M_p(F(x_j))].$$

In a similar way, we can group objects in X with respect to any set of meta-actions, and in particular with respect to a minimal set of meta-actions $\{M_p\}_{p \in P}$ triggering a given action rule $r(r_1, r_2)$. Let us assume that $G = \{G_k\}_{k \in K}$ represents that grouping. Next, we split each group $G_k \in G$ in such a way that objects having the same side effects with respect to meta-actions associated with G_k are placed in the same sub-groups. This strategy is similar to the one presented in section 4.1.

5.3.2 Experiments

In this section, we used R 2.15 to process the data and a built-in software to discover action rules.

5.3.2.1 Side Effects Based Grouping

In this experiment we used the steps described in our proposed approach. However, we first cleaned the data and organized it by negative side effects. Since we already know the three side effects in our dataset, we grouped patients that have the same negative side effects by codifying the different combinations of side effects values. Since we have three side effects and two possible values for each side effect (0 for negative and 1 for positive), we will have eight (23=8) possible groups of patients (000, 001, 010,. . . , 111) in our partitioning. Grouping patients with regards to same side effects resulted in the number of patients depicted in Table 12 in each group:

Table 12: Number of patients by side effects groups.

Groups	g_{000}	g_{001}	g_{010}	g_{011}	g_{100}	g_{101}	g_{110}	g_{111}
# Patients	46	38	17	54	18	44	32	262

After grouping the patients by side effects, we extracted the action rules $AR(2, 85\%)$ using our action rules discovery software. We constrained the action rules confidence to 85% and support to a minimum of 2 in each one of the groups. However, since different groups might have different support, we increased the support sequentially to have the highest minimum support that returns actions rules. Furthermore, we fixed the decision transition from no-improvement to improvement $(score, 0 \rightarrow 1)$. The results for each group of action rules discovery after the partitioning are presented in Table 13:

Table 13: Action rules by side effects groups.

Groups	Support	Confidence	# Rules
g_{000}	-	-	0
g_{001}	2	100	251
g_{010}	2	100	8128
g_{011}	5	100	3
	5	86	1
g_{100}	2	100	152213
g_{101}	3	100	720
g_{110}	3	100	5
g_{111}	-	-	0

Note that the group of only negative side effects g_{000} and the group of only positive side effects g_{111} does not have any action rule. This is due to the decision feature being the same for all the group. In addition, note that the extracted rules do not have high support. This is due to shrinking the patients populations in the groups. An example of an action rule extracted from group g_{110} is shown in the following: $(F.24, 4 \rightarrow 4) \wedge (C.8, 4 \rightarrow 4) \wedge (F.7, 4 \rightarrow 2) \wedge (C.19, 4 \rightarrow 4) \wedge (E.10, 4 \rightarrow 2) \Rightarrow (ScT, 0 \rightarrow 1), sup = 3, conf = 100\%$.

5.3.2.2 Action Rules Based Grouping

After the data cleaning step, we extracted all possible action rules from the entire decision system. We used our action rules extraction software setting up the minimal confidence to 85%, and the starting support at a minimum of 20. This support was then decreased sequentially until we reached a minimum support that resulted in discovering at least one action rule. First, we extracted two action rules with minimum support 20 and minimum confidence 85%. Then, we decreased the minimum support to 19 to extract more action rules while keeping the confidence at least at 85%. This way 10 additional action rules has been found. Next, we grouped them into sets of action rules that have the same antecedent side. Grouping the action rule with same antecedent side resulted in two groups for the first partition P_1 for action rules with support 20. For the second set of action rules with support 19, we ended up having a partition P_2 of 10 groups. Each one of those groups is summarized in Table 14 and contains one action rule with a specific confidence and support:

For each action rule antecedent side set, we grouped patients that have the same preconditions as the antecedent part of action rules set together in the same group. This type of partitioning is natural since each patient $x \in X$ is associated with a set of possible action rules $AR_S(x)$. This step also insures that patients that do not have possible action rules $AR_S(x) = \emptyset$ are not part of the partition grouping.

Each group of action rules led to a number of patients having the same preconditions as the antecedent side of the rules. This experiment returned the same number of patients in each group as the support of the respective rules, which is 20 patients

Table 14: Action rules by antecedent side grouping.

Partitions	Groups	Support	Confidence	# Rules
P1	G1	20	86.95	1
	G2	20	87.41	1
P2	G1	19	86.36	1
	G2	19	85.58	1
	G3	19	87.73	1
	G4	19	87.55	1
	G5	19	86.36	1
	G6	19	86.70	1
	G7	19	86.85	1
	G8	19	86.36	1
	G9	19	86.99	1
	G10	19	86.85	1

for the two groups of P_1 and 19 patient for each group of P_2. Note that we do not get the same group size as the support of the corresponding rules in all situations since we are using the minimum support method to compute the rule support. We followed the next step in our described approach where we further partitioned each group of patients presented in Table 14 to subgroups having the same negative side effects. Each action rules based group was partitioned into a number of subgroups with regards to the same negative side effects. This partitioning is represented in the following Table 15, where you can note that the total number of patients for all negative side effects sub-group (row sum) is larger than the total number of patients in each corresponding parent group in the action rules grouping. This is due to an overlap between the groups for patients having different applicable action rules.

Table 15 also represents the merging step, where the total number of patients in each sub-group is represented by the partition with respect to side effects. Note that this number is small due to the patients overlap described earlier in the table for the

Table 15: Number of patients by side effects and action rules grouping.

Partition	Group	g_{000}	g_{001}	g_{010}	g_{011}	g_{100}	g_{101}	g_{110}	g_{111}
P 1	G1	8	2	2	4	1	1	1	1
	G2	8	6	3	2	0	1	0	0
Total P1		10	8	4	4	1	2	1	1
P 2	G1	7	2	1	4	1	1	2	1
	G2	7	2	1	4	1	1	2	1
	G3	8	5	2	3	0	1	0	0
	G4	7	5	2	3	0	1	1	0
	G5	8	5	2	3	1	0	0	0
	G6	7	5	3	3	0	1	0	0
	G7	8	5	2	2	0	1	1	0
	G8	9	5	1	3	0	0	1	0
	G9	8	5	2	3	0	1	0	0
	G10	8	5	2	3	0	1	0	0
Total P2		9	6	3	6	2	2	2	1

action rules groups. However there is no overlap between the different side effects groupings.

5.3.2.3 Meta-action Based Grouping

This experiment requires grouping the objects based on meta-actions. Since the physician already applied treatments (meta-actions) to patients and those treatments are recorded in the dataset, we just need to start grouping patients in the same group if the same treatments have been applied to them. After cleaning the dataset and removing the missing and incomplete values, we ended up having three possible treatments that are Hearing Aid (HA), sound Generators (SG), and Regular Consultation (RC). Thus, we grouped the patients into those three different groups. The results of the grouping based on same meta-actions are summarized in Table 16.

We then generated action rules from each group by setting up the minimum con-

Table 16: Number of patients by meta-action.

Group	HA	SG	RC
# Patients	16	87	414

fidence to 85% and minimum support to 3; however, the support varies from one group to another depending on the groups action rules strength. We also fixed the decision feature transition from 0 to 1 (no improvement to improvement in the score $(ScT, 0 \to 1)$. Table 17 summarizes the results of action rules discovery:

Table 17: Action rules by meta-actions groups.

Group	Support	AVG Confidence	# Rules
HA	3	100	16
SG	4	100	16
RC	21	87.5	1
	19	87.56	7

We can note that the minimum support strength of the discovered action rules in each group is positively correlated to the number of patients in each group. Here is the example of an action rule discovered in the RC group:

$(g, 3 \to 3) \wedge (C.8, 4 \to 0) \wedge (F.7, 4 \to 0) \wedge (C.19, 4 \to 0) \wedge (E.10, 4 \to 0) \wedge (E.17, 4 \to 0) \wedge (E.16, 4 \to 0) \Rightarrow (0 \to 1),\ sup = 15,\ conf = 100\%$

Once we have the groups of patients based on the meta-actions, we further partition each group to sub-groups with respect to the same negative side effects. The results are summarized in Table 18.

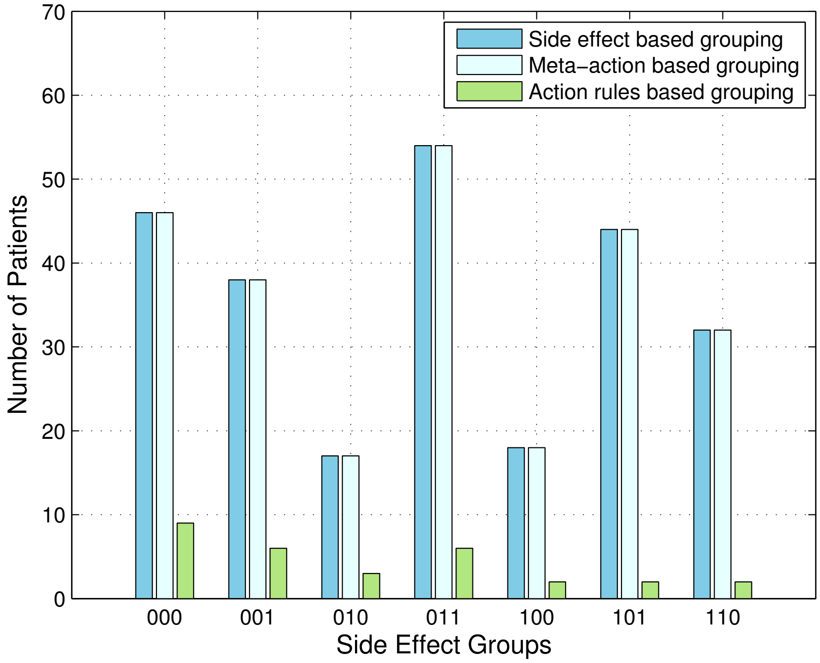


Figure 8: Patient population by side effect groups.

Table 18: Number of patients by meta-actions and side effects grouping.

Groups	g_{000}	g_{001}	g_{010}	g_{011}	g_{100}	g_{101}	g_{110}	g_{111}
HA	2	0	0	2	1	2	1	8
SG	9	6	0	10	4	7	1	50
RC	35	32	17	42	13	35	30	210

5.3.2.4 Grouping Schemes Comparison

All three grouping schemes have their advantages and disadvantages with regards to action rules personalization. The side effects-based grouping is a patients-centric scheme with regards to their negative side effects. It allows the extraction of more personalized action rules since each group dataset used in the action rules discovery process is exclusive to patients with exactly the same negative side effects. As we can note from Table 12 and Table 13, and Figure 8, the average group size is relatively

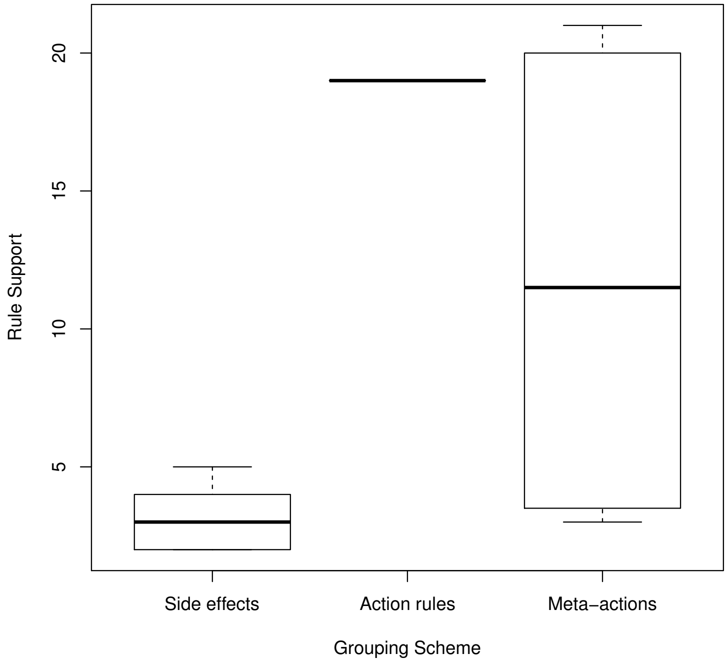


Figure 9: Support of each grouping scheme.

high and the number of action rules generated is substantially high. In addition Figure 10 denotes the highest confidence for this scheme. However the support of the extracted rules is rather small in comparison with the other schemas as seen in Figure 9. In fact, using the partitioned datasets to extract the rules limits the number of observations; thus, limits the strength of the action rules. In this sense, we might argue that the second scheme, action rules based grouping, is more efficient than the previous schemes. It uses the whole dataset to generate the action rules first, and then groups patients by their side effects. It allows extracting action rules with stronger support as seen in Figure 9 and Table 14. This partitioning is the most fine-grained since it allows distinguishing the patients not only by their side effects but also by their personalized rules. This is confirmed in our experiment where the

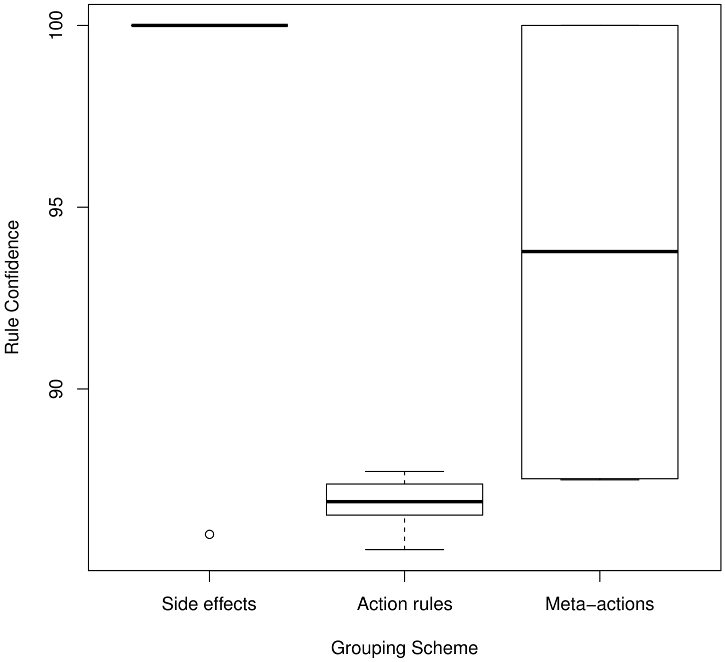


Figure 10: Confidence of each grouping scheme.

number of patients in the fine-grained side effects groups is very small as seen in Figure 8 and Table 15. This is due to shrinking the datasets or groups to only the action rules domains (patients with extracted applicable action rules $AR_S(x)$). The total number of patients will get bigger as we decrease the support, since we will extract more rules. However, the rules confidence is the smallest in this scheme as seen in Figure 10. In addition, in order to cover all patients in the dataset, we will eventually have to decrease the support; thus, the strength of the rules. The third grouping scheme, meta-action based grouping, is the most efficient for our dataset. As we can note from Table 17 and Figure 9, even though we used only subsets of the overall dataset in the meta-actions based grouping, we extracted a larger number of rules with the highest support for the RC group in the later scheme. Furthermore, extracted rules have a higher average confidence as seen in Figure 10 than in the second scheme. In addition, the average number of patients in each side effects based sub-groups is larger than in the previous scheme and encompasses all patients for our dataset as seen in Figure 8. Grouping by meta-actions filtered all the noise, such as, patients that have the same initial state but different decision feature values than the ones applied in the extracted rules.

In this section, we studied the tinnitus handicap disease, and noted the importance of filtering negative side effects when applying treatments to patients. We then formally defined the negative side effects, and proposed to extract personalized action rules based on these side effects. We believe that expert physicians partition patients with the same pathological state based on their side effects response to treatments when they are available. Therefore, we proposed three grouping schemes with regards to negative side effects to extract patients personalized action rules. We also implemented the three grouping schemes and tested them on the Tinnitus handicap dataset. We further compared the three grouping schemes and discussed their advantages and disadvantages, and justified our choice to apply the meta-action based grouping scheme. We trust that personalization is a very important aspect in filtering noise that skilled experts face when making decisions.

5.4 Chapter Conclusion

Actions rules are very important in modeling expert and domain knowledge. They were augmented by the introduction of meta-actions that experts use to trigger them. In this chapter, we analyzed side-effects and mined negative action sets resulting from the application of meta-actions. We also proposed treatments' negative side-effects prediction technique for patients based on patients' clusters. Furthermore, we presented three patient's grouping schemes based on available side-effects and compared them. The results of the evaluation and comparison showed that meta-actions based grouping scheme is the best grouping scheme for personalized action rules extraction.

CHAPTER 6: HEALTH CARE META-ACTIONS IN PLAY

There are several applications of meta-actions' effects in the healthcare domain as well as other domains. Meta-actions play an important role in representing the actionable knowledge that practitioners need. In this chapter, we present some of the applications of meta-actions' effects in healthcare systems.

We first start by using meta-actions' effects to reduce action rules that are not executable in their current state and present dangerous side effects. This is done by analyzing the execution of the action rule at hand, and selecting the meta-actions that help trigger this rule. In addition, we study patient's side effects that the selected meta-actions can introduce on action rules. These side effects are outside of the action rule scope but in the patient's precondition. A utility weighted sum is then introduced to balance the execution of action rules and reduce side effects ratio.

We then present new action rules evaluation metrics based on meta-actions and demonstrate their usefulness. We used the likelihood and execution confidence to measure the action rules robustness. Furthermore, we used a cost and satisfaction model to evaluate the application of meta-actions.

The two applications of meta-actions presented in this chapter are not isolated cases, but they are good examples on how meta-actions can be applied and offer a good perspective for future work.

6.1 Personalized Action Rules Reduction

Action rules model the correlation between the transitions of each objects' attribute values, and the decision attribute values transitions. They are closely related to meta-actions, that constitute the main trigger for their antecedent side transitions. For this reason, it is important to extract action rules and meta-actions and study their connection in an integrated way. To trigger an action rule, we need to find the set of meta-actions that cover the whole antecedent side of the action rule. Another important factor of action rules execution is the number of object's side effects introduced by meta-actions used. Existing research has proposed the lookup of minimum number of meta-actions covering an action rule while minimizing the side-effects.

In this chapter, we propose an action rules reduction mechanism [32] to reduce the meta-actions used, and therefore, reduce the number of side effects introduced. The reduction process consists of substituting the action rule atomic action terms as summarized in Figure 11. This process is done by removing the uncovered atomic action terms and replacing them by highly correlated action terms that are covered by meta-actions action terms. Similarly, if we start with the meta-actions, an action rule could be obtained by composing the necessary meta-actions action terms that are correlated with the desired action rule decision.

6.1.1 Action Rules Substitution

Let us assume that we have a system $S = (X, F \cup \{d\}, V)$, where X is a set of objects, $F = \{a, b, c, e, f\}$ is the set of attributes, and $V = \{a_1, a_2, b_1, b_2, c_1, c_2, e_1, e_2, f_1, f_2, d_1, d_2\}$ is the domain of attribute values. If we have an action rule of the

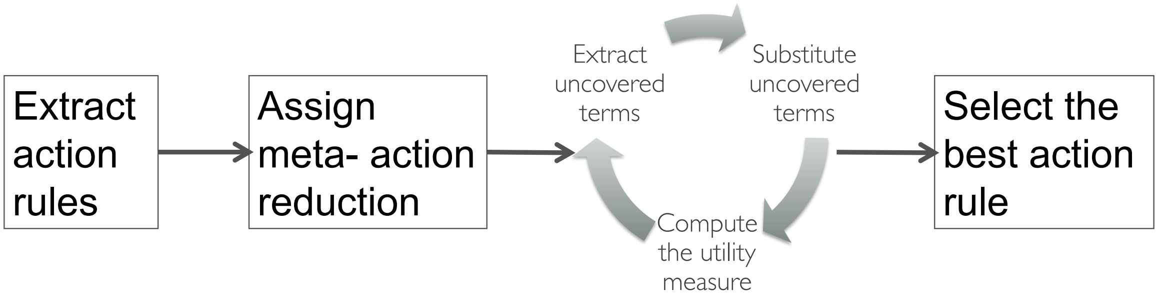

Extract action rules
Assign meta- action reduction
Extract uncovered terms
Substitute uncovered terms
Compute the utility measure
Select the best action rule

Figure 11: Action rules reduction methodology.

form $r = [((a, a_1 \to a_2) \wedge (b, b_1 \to b_2) \wedge (c, c_1 \to c_2)) \Rightarrow (d, d_1 \to d_2)]$, an object $x \in X$ such that $F(x) = \{a_1, b_1, c_1, d_1, e_1, f_1\}$, and two meta-actions m_1, m_2 where $m_1 = \{(a, a_1 \to a_2), (b, b_1 \to b_2), (e, e_1 \to e_2), (f, f_2 \to f_1)\}$, and $m_2 = \{(a, a_1 \to a_2), (c, c_1 \to c_2), (e, e_1 \to e_2), (f, f_1 \to f_2)\}$, we can simply use both meta-actions m_1 and m_2 to trigger the rule r. We can also reduce the action rules to use only one of the two meta-actions by computing the correlation between each meta-action action terms and the uncovered action rule atomic action terms. This way, if we use the meta-action m_1, we obtain the set of uncovered atomic action terms by the rule r, which will be the set $\{(c, c_1 \to c_2)\}$. Next we can compute the possible correlated action terms between the meta-action and the uncovered set. For example, the action term $[(a, a_1 \to a_2) \wedge (e, e_1 \to e_2)]$ might be correlated with the uncovered action term $(c, c_1 \to c_2)$ for m_1. Similarly, if we use m_2, we obtain a different set of uncovered action terms which will be $(b, b_1 \to b_2)$, and for instance, if the action term $(f, f_1 \to f_2)$ is highly correlated with $(b, b_1 \to b_2)$, we can substitute $(b, b_1 \to b_2)$ with $(f, f_1 \to f_2)$ and use only the meta-action m_2.

Note that we only use action terms of size one (atomic action term) from the uncovered set of action terms; however, we may take action terms of any size from the meta-actions to compute the two action terms correlation. This correlation is computed using action rules confidence, which is computed by modeling an action rule based on the uncovered atomic action terms representing the decision transition, and the meta-action action terms representing the antecedent side of the action rule. This way, we can compute the confidence by substituting any uncovered action term by any available action term in the meta-action or set of meta-actions. This method will result in a large number of possible substitutions with their respective confidence. To this extent, we can select action terms triggered by meta-actions with higher confidence for each uncovered atomic action term to be applied to model the new action rule. However, we would like to optimize the number of meta-actions used, or else we could have used the naive solution to cover all the action rule antecedent side. In addition, we aim to minimize the number of side effects resulting from applying the meta-actions. We can build an optimization function based on the maximization of the utility O for each meta-action m and its atomic action terms $t_j \in m$ covering both the possible terms of action rule r_i and each uncovered atomic action term t_u. The utility O will minimize the number of side effects and maximize the confidence of the new generated rule r_i (action terms correlation) such that:

$$O(m, r_i) = \beta \cdot Conf(r_i) \cdot \prod_{j=0}^{n} [Conf(t_j) + (1 - \beta) \cdot \tfrac{|t_j|}{|m(x)|}],$$

where β is a diffusion coefficient, $Conf(r_i)$ is the confidence of the new generated rule from the action terms t_j and t_u, and $Conf(t_j)$ is the confidence of the coverings of the uncovered set. The value of $Ratio(t_j) = \frac{|t_j|}{|m(x)|}$ represents the ratio between the negative side effects and the atomic terms used. It is important to note that the ratio of the side effects $Ratio(t_j)$ is fixed for any given action term extracted from the same meta-action. Therefore, when using only one meta-action only $Conf(r_i)$ will help us decide which action term to choose for the substitution. On the other hand, if we have multiple meta-actions involved, it is important to take into consideration the number of side effects. We also define an execution confidence metric that models the execution confidence of meta-actions triggering an action rule and reaching the desired result. We derive the execution confidence metric from the right hand side of the utility formula as follows:

$$ExConf_{i,j} = \prod_{i=0}^{m} Conf(r_i) \cdot \prod_{j=0}^{n} Conf(t_j)$$

Where m is the number of action rules involved, such as the original rule or the substitution rules, and n is the number of action terms covered by the meta-actions used.

6.1.1.1 Intra Meta-action Composition

The composition of meta-action action terms can happen within each meta action, which means that we create new action terms based on the already existing candidate action terms that are different. This process is pretty tedious since it requires the combination of all possible atomic action terms extracted from our data, and verifying that they respect the action term definition introduced in the Preliminaries section. Algorithm 1 describes the method used to extract all the possible action terms.

In this algorithm, we initialized the set A_1 by the atomic action terms that are action terms by definition. Then, for each possible action term size, we generated the candidate action term C_{k+1} by composing all the previous candidate action terms in

C_k with atomic action terms in A_1. The new candidate action terms of size $k+1$ were then evaluated and inserted in the action terms A_{k+1} of size $k+1$ if they satisfied the action term definition.

6.1.1.2 Inter Meta-actions Composition

Inter meta-action composition of action terms is used when the use of more than one meta-action is necessary. It is more complex than intra meta-action composition as it involves duplicates of action terms and additional negative side effects. We can use the previous algorithm to generate the new action terms from multiple meta-actions since the candidate action terms $c_k \in C_k$ are defined in sets which will ignore the duplicates action terms. However, it is important to keep track of the action terms generated from each meta-action, as well as from multiple meta-actions. Therefore, we slightly modified the previous algorithm to fit multiple meta-actions. Algorithm 2 describes the method used to extract all the possible action terms from multiple meta-actions.

Algorithm 2 is used to generate action terms for two meta-actions sets M_i and M_j by generating all possible compositions of their respective action term sets. If one of the meta-action sets, M_i for instance, contains only one meta-action, we use Algorithm 1 to generate its respective action terms set A_i, and if the meta-action set contains two or more meta-actions, we use Algorithm 2 to recursively generate their respective action terms set.

```
Data: $A_k$ : Set of frequent action terms of size $k$
$C_k$ : Set of candidate action term of size $k$
Result: $\{A_k : k \geq 1 \;\&\; card(A_k) \geq 1\}$
$A_1 :=$ {frequent atomic action terms}; $k := 1$ ;
while $card(A_k) \geq 1$ do
    $C_{k+1} :=$ new candidates generated from the combination of $A_k$ and $A_1$;
    for for each transaction pair $p \in P$ extracted from the database do
        increment the count of all candidates in $C_{k+1}$ that are contained in $p$ ;
        for $c_{k+1} \in C_{k+1}$ do
            if $c_{k+1} = c_k \wedge a_1$ && for any two atomic action terms
            $(f, v_1 \rightarrow v_2), (g, w_1 \rightarrow w_2)$ contained in $c_{k+1}$ we have $f \neq g)$ then
                $A_{k+1} := A_{k+1} \cup c_{k+1}$ ;
            end
        end
        $k := k + 1$;
    end
end
```

Algorithm 1: Generating Intra Meta-actions Action Terms.

6.1.2 Experiments and Evaluation

This section describes the experimental setting, the data set used and the different results obtained. In the following experiments, we have assumed that we already extracted a set of action rules.

6.1.2.1 Action Rules Reduction Evaluation

In this section, we describe the method used to create new action rules that substitute the uncovered set of atomic action terms. We implemented a Java program that reads the different actions terms and computes their likelihood and confidence. We also generated the uncovered set to create the substitution action rules based on all the possible meta-actions action terms, and computed their confidence. We then selected the best substitution action rules to apply using their confidence and utility described earlier in the action rules reduction. The rest of the experiments assume

```
Data: $A$ : Set of sets of frequent action terms $A_i$ for each meta action $M_i$
$G_{i,j}$ : Set of frequent action terms for two meta actions $M_i$ and $M_j$
$C$ : Set of candidate action term of size $k$
Result: $G_{i,j}$ : $card(G_{i,j}) \geq 1$
$A_i$ := {frequent action terms of the meta-action $m_i$}; $A_j$ := {frequent action
terms of the meta-action $m_j$}; $C$ := new candidates generated from the
combination of one action term from $A_i$ and one from $A_j$;
for for each transaction pair $p \in P$ extracted from the database do
    Increment all candidates C that are contained in P to be able to use them
    for $c \in C$ do
        if $c = a_i \wedge a_j$ && for any two atomic action terms
        $(f, v_1 \rightarrow v_2), (g, w_1 \rightarrow w_2)$ contained in c we have $f \neq g$) then
            $G_{i,j} := G_{i,j} \cup c$ ;
        end
    end
end
```

Algorithm 2: Generating Inter Meta-actions Action Terms.

that the diffusion coefficient $\beta = 0.5$; however, it is up to the practitioner to select which way the bias should be (side-effects, or execution confidence).

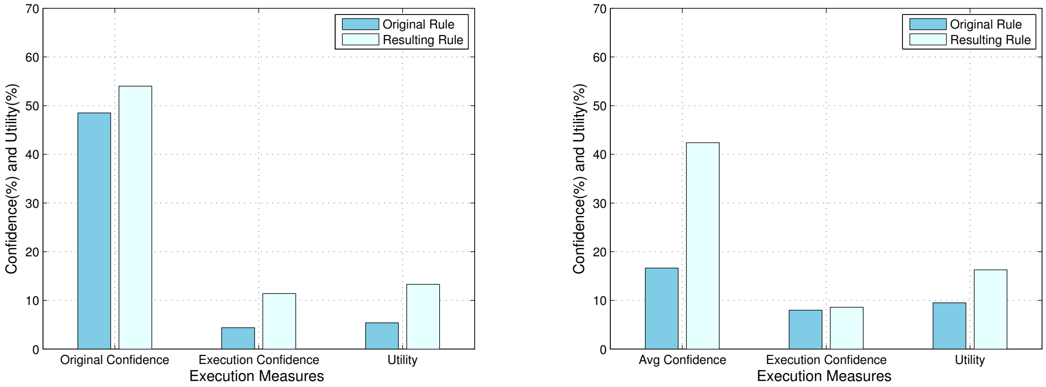
We chose one action rule with a confidence of 100% and we ran our program with this rule $[(C.11, 4 \rightarrow 4), (F.15, 2 \rightarrow 4), (F.7, 4 \rightarrow 2) \Rightarrow (class, 0 \rightarrow 1)]$. When we used the meta-action RC, all the antecedent side of the action rule r was covered with an execution confidence of 5.16%, thus there is no need to consider reducing it. However, when we used the meta-action SG, our program returned that the action term $[(F.15, 2 \rightarrow 4)]$ was not covered with this meta-action. Therefore, applying this meta-action will result in executing the rule $r = [(C.11, 4 \rightarrow 4), (F.7, 4 \rightarrow 2) \Rightarrow (class, 0 \rightarrow 1)]$. For the meta-action SG, our reductions mechanism will reduce the original action rule to an equivalent rule $r_r = [(C.11, 4 \rightarrow 4), (F.7, 4 \rightarrow 2), t_j \Rightarrow (class, 0 \rightarrow 1)]$, where t_j is the action term providing the best confidence for the substitution rule $r_s = [t_j \Rightarrow (F.15, 2 \rightarrow 4)]$. For this meta-action, we found that $t_j =$

$[(C.11, 2 \rightarrow 4), (F.13, 4 \rightarrow 4)]$ provides the best confidence out of all the combinations of action terms in SG. When we used the meta-action HA, we found the uncovered set $[(C.11, 4 \rightarrow 4), (F.7, 4 \rightarrow 2)]$ that will trigger only the rule $r = [(F.15, 2 \rightarrow 4) \Rightarrow (class, 0 \rightarrow 1)]$. Similarly, the original rule is reduced to the equivalent rule $r_r = [(F.15, 2 \rightarrow 4), t_i, t_i \Rightarrow (class, 0 \rightarrow 1)]$ with respect to t_i and t_j that are the action terms substituting the two atomic terms in the uncovered set. In addition, you will notice that any two combinations of our three meta-actions will cover the whole antecedent side of the original rule. Table 3 compares between the generated action rules in terms of average Rules Confidence, Execution Confidence, and Utility O.

Table 19: Tinnitus action rules reduction statistics.

Action terms → *Meta−actions* ↓	Action Rule	Avg Confidence	Execution Confidence	Utility O
SG	r	48.5	4.4	5.4
	r_s	54	11.1	13.3
HA	r	16.65	8	9.5
	r_s	42.4	8.6	16.29
RC	r	100	5.16	6.4

A practitioner would be more inclined to use the meta-action RC since it covers all the antecedent side of the original rule and its original confidence is of 100%. However, Table 19 shows that the execution confidence of using SG is 11.1% or HA is 8.6% with the rules substitutions is in both cases higher than the execution confidence of RC which is 5.16%. Furthermore, the respective utilities of SG and HA are also higher then RC. Table 19 also shows that the average confidence of the substitution rules r_s based on both meta-actions SG and HA are higher than the confidence of r. In addition, Figure 12 shows that the execution measures of the resulting substitution rules are larger then the ones for the original rule in terms of all average rule confidence, execution confidence, and utility. Therefore, using r_s would lead to a more probable execution of the original action rule. Furthermore, the substitution rule r_s models the correlation between the original rule r and r_r, thus leading to a cascading effect from the rule r_r to r.



(a) Action rules comparison for meta-action SG (b) Action rules comparison for meta-action HA

Figure 12: Comparison of the execution measures for both SG and HA.

In previous work, action rules were discovered and selected based on their support and confidence regardless of their executability. Meta-actions were then selected to trigger the action rules without thorough examination of whether there will be negative side effects or not. In this chapter, we looked at the problem of action rules discovery from all angles. We proposed an action rules reduction process that helped us to refine action rules based on their meta-actions. We implemented our approach on the tinnitus handicap data, and evaluated our action rules reduction process. We did not evaluate our mechanism's execution time performance, which will increase with increasing the size of the data and the number of possible action terms combinations. Our experiments show that using meta-actions in the process of reduction and substitution of action rules increases the likelihood and execution confidence results for the action rules. Furthermore, we particularly included the side effects in the utility optimization process to reduce the negative side effects based on their correlation with the antecedent side of action rules.

6.2 Meta-action as a Tool for Action Rules Evaluation

Action rules extraction is a field of Data Mining used to extract actionable patterns from large datasets. Action rules present users with a set of actionable tasks to follow in order to achieve a desired result. An action rule can be seen as two patterns of feature values (classification rules) occurring together and having the same features. They are evaluated using their supporting patterns occurrences in a measure called *Support*. They are also evaluated using their *Confidence* that represents the product probability of the two patterns confidences. These two measures are important to evaluate action rules; nonetheless, they fail to measure the features values transition correlation and applicability. The core of the action rules extraction process that extracts independent patterns and constructs an action rule is the reason for failing to measure the patterns correlation. In this chapter, we present the benefits of meta-actions in evaluating action rules in terms of two measures, namely *Likelihood* and *Execution Confidence*. In fact, in meta-actions, we extract real features values transition patterns, rather than a composing two feature values patterns. We also present an evaluation model of the application of meta-actions based on *Cost* and *Satisfaction*. We extracted action rules and meta-actions and evaluated them on the Florida State Inpatient Databases (SID) that is part of the Healthcare Cost and Utilization Project (HCUP) to evaluate our methodology.

6.2.1 Meta-Actions Versus Action Rules

Meta-actions and action rules are similar concepts since they aim at extracting transition patterns from information systems. However, the meta-action extraction process extracts a subset of the action terms extracted by the action rule extraction process. The similarities and differences of both concepts are discussed in this section and reported in Table 20.

Action rules are commonly used by decision makers to discover possible changes in objects' state, which would ultimately transition the objects' overall state to a more desirable state with regards to the decision feature. There exist several techniques to extract action rules. The datasets used to extract action rules are composed of instances, where each object is described by one instance. In other words, each object is recorded once in the dataset, and the object state is represented by its instance features values. All possible transitions from any instance to any other instance in the dataset are discovered in the action rules discovery process regardless of the order in which they were recorded. This suggests that there is no temporal ordering relation amongst instances. Therefore, some transitions discovered in the action rules discovery process may not be applicable in a real life scenario. For example, let us assume an information system that models cancer patient visits, where each patient visit describes the patient pathological state in term of diagnoses. In this example dataset, each patient visit represents an instance. Now, let us assume that each visit record includes three diagnoses as features, namely: tumor size, Chemo level, and number of chemotherapy performed. Also, it includes the information that indicates whether the patient is stable or unstable, stored as decision feature at the visit time. The diagnosis features are recorded as $TSize$ for tumor size, $CLevel$ for chemo level and $NChemo$ for the number of chemotherapy already performed on the patient, and the patient overall state recorded as $State$. Note here that each patient visit is considered as an object for the action rule discovery process. The following action rule may be discovered by the system: $(TSize,\ 7 \rightarrow 4) \wedge (CLevel,\ 4 \rightarrow 3) \wedge (NChemo,\ 5 \rightarrow 4) \Rightarrow (State,\ Unstable \rightarrow Stable)$. In this action rule, we are interested in the patient overall state $State$, and we would like to move patients from an unstable state to a stable one. Nonetheless, we cannot transition the $NChemo$ from 5 to 4 since we already performed 5 chemo on this patient. Therefore, this action rule would not be applicable in real life. It was discovered using all the visits in our dataset, where each visit is considered as an object, regardless of the fact that a patient may have several visits to the hospital. In other words, patients are not taken into consideration since the object instances are visits and not patients.

Meta-actions on the other hand, represent transitions from instances that occurred in a specific order for specific objects in real life scenarios, which insures their applicability. Meta-actions do not model rules, they rather model transition effects that are represented in an influence matrix. By using the previous example and adding a new feature that describes the visit number, we could sort the visits by their temporal order. In addition, we treat our system as multiple subsystems of visits identified by the patient ID. This way, we could mine real transitions in terms of patients pathological state. As a result the transition $(NChemo,\ 5 \rightarrow 4)$ would not be extracted since the chemo number 4 would have taken place before chemo number 5, and the meta-action mining process takes the visit order into consideration. In addition, different patients might react differently to the chemotherapies with regards to their cancer level and tumor size. Using meta-actions, we extract transitions that occurred in real life for each patient. For instance, we may extract the following transitions:

$(TSize,\ 7 \rightarrow 6) \wedge (CLevel,\ 4) \wedge (NChemo,\ 5 \rightarrow 6) \wedge (State,\ Unstable \rightarrow Stable)$, where the left hand side of the action term took place at a visit that occurred prior to the right hand side. Furthermore, the transitions were mined from instances belonging to unique patients. In summary, the introduction of the instances order and their identifier with regard to the object eliminated any confusion.

Table 20: Meta-actions vs. action rules.

Action Rules	**Meta-actions**
One object is one instance	One object is a set of instances
No instance order	Temporal instance order
Support is minimum support of classification rule	Likelihood is the number of real transitions
Confidence independent probability	Term confidence and execution condidence
Possible transitions	Real transitions
Rules	Collection of action terms

The purpose of meta-actions is to trigger transitions in feature values that will change the object state. Eventually, the transitions triggered by meta-actions will trigger an action rule and will cascade into transitioning the decision feature value. As mentioned earlier, meta-actions' transitions are a subset of the transitions extracted in the action rules extraction process. In addition, meta-actions play a passive role, in the sense that they do not change decision feature values. They rather inform decision makers of possible transitions. On the contrary, action rules play an active role that help decision makers drive their strategy, and explicitly look into the transitions that affect the decision feature. In other words, meta-actions are not replacing action rules; instead, they are enhancing the process of selecting the best amongst them.

6.2.1.1 Action Rules Selection by Meta-actions

In the effort of selecting the best action rules, meta-actions play an important role. After extracting action rules, meta-actions inform us, amongst other things, on whether the extracted action rules are applicable or not. Meta-actions also provide decision makers with the confidence of executing the antecedent side of an action rule. Given a set of action rules, and an influence matrix describing meta-actions transitions in our system, we strive to select the best applicable action rules and their respective triggers. This is done by first dividing the set of action rules into applicable and not applicable action rules. Applicable action rules are the rules, whose antecedent sides are covered by the influence matrix. Then, for each action rule, we select the best coverings of the action rule from the influence matrix. The best coverings of an action rule are the maximal action terms in the influence matrix. By maximal action term, we mean the action term with the highest number of atomic terms covering the action rule. This is performed by intersecting the antecedent side of the action rule with all the action terms in the influence matrix. It is important to note that the action terms with higher number of atomic action terms will have a higher confidence. As one can guess, two or more action terms maybe required to cover an action rule. Similarly, two or more meta-actions may be required to cover an action rule.

We described earlier, how to compute the confidence of an action term $TermConf$ in Chapter 4; however, we still need to define how to compute the confidence of multiple action terms, namely global confidence $GlobConf$. Computing the confidence of multiple action terms depends on whether the action terms belong to the same meta-action or not. In fact, action terms $\{t_1, t_2, \ldots, t_n\}$ that belong to different meta-actions are independent since they are extracted from different object populations; therefore, their confidences are independent and their global confidence is the product probability of independent confidences, as shown in the following:

$$GlobConf(\{t_1, t_2, t_3, \ldots, t_n\}) = \prod_{i=1}^{n} TermConf(t_i). \tag{14}$$

On the contrary, action terms that belong to the same meta-action are extracted from the same object population, and might be extracted from the same objects. Therefore, such action terms are dependent on the objects, and their global confidence is dependent on their transition probability. To avoid confusion, and for the sake of simplicity, we will define the global confidence of action terms from the same meta-action as follow:

$$GlobConf(\{t_1, t_2, t_3, \ldots, t_n\}) = Like(\bigcup_{i=1}^{n}\{t_i\})/sup(Left(\bigcup_{i=1}^{n}\{t_i\})) \tag{15}$$

Global confidence will inform us on how well we can trigger the antecedent side of an action rule; however, it does not inform us about how well the features values transitioned by the meta-action will cascade into the desired decision feature value. For this reason, we define a new metric called execution confidence $ExConf$ that computes the confidence of execution for an action rule. The execution confidence of an action rule r triggered by the set of meta-actions m is defined as follow:

$$ExConf(m, r) = GlobConf(m(r)) \cdot Conf(r) \tag{16}$$

where $m(r)$ represents the set of action terms used in triggering the antecedent side of the action rule r.

With the introduction of the $ExConf$, one could select the action rules with the highest execution confidence, along with their corresponding meta-actions. Doing so would ensure that the action rules chosen are more likely to be accurate. However, decision makers may also be interested in the highest return on investment solution, which would be good enough to solve the issue in hand. In fact, meta-actions are commonly associated with a cost based on the domain of interest. For example, in the healthcare domain, each treatment is associated with a cost, and patients are discharged with a bill including all their medical expenses.

Let us assume that cost C_i is associated with each meta-action $m_i \in M(S)$ used. Then, a good measure associating the cost C_i and the execution confidence $ExConf$ would be the satisfaction rate noted as $SatRate$. The satisfaction rate gives a pointer to which action rule r is good enough; in other words, which action rule and corresponding meta-actions incurs the minimum cost while returning an acceptable execution confidence. The satisfaction rate for a rule r and a set of corresponding meta-actions M_r is computed as follow:

$$SatRate(M_r, r) = \frac{ExConf(M_r, r)}{\lambda \sum_{i=1}^{n} C_i} \tag{17}$$

where n is the number of meta-actions used, and $0 < \lambda \leq 1$ is a cost coefficient chosen by decision makers.

Table 21: Description of CCS single level procedure codes.

PRCCS1	**Description**
43	Heart valve procedures
44	Coronary artery bypass graft (CABG)
45	Percutaneous transluminal coronary angioplasty (PTCA)
47	Operations on lymphatic system
48	Insertion; revision, replacement; removal of cardiac pacemaker or cardioverter/ defibrillator

6.2.2 Experiments

We performed a set of experiments on the Florida State Inpatient Database using our meta-action and association action rules extraction system. We finally evaluated the action rules extracted using the meta-actions applied.

6.2.2.1 Meta-action Extraction

We extracted meta-actions action terms from the Florida SID dataset, and computed their likelihood and term confidence. In these experiments, we extracted action terms for 231 procedures, considered here as meta-actions; however, for the sake of this chapter, we report the results for five meta-actions described in Table 21 with their CCS procedure codes. We built an influence matrix like table shown in Table 23 that describes our findings in terms of meta-actions' action terms. Table 23 reports few example of action terms of size 1 to 4 extracted for each meta-action; however, we extracted action terms with more than 4 atomic terms and we may extract action terms of size up to the features number. We also included the description of the CCS diagnoses codes for some of the most significant action terms' features in Table 22.

As mentioned in the dataset description, the flexible attributes represented by

Table 22: Description of CCS single level diagnosis codes [4].

Diagnosis code	Description
101	Coronary atherosclerosis and other heart disease
55	Fluid and electrolyte disorders
62	Coagulation and hemorrhagic disorders
106	Cardiac dysrhythmias
99	Hypertension with complications and secondary hypertension
158	Chronic kidney disease
114	Peripheral and visceral atherosclerosis
108	Congestive heart failure; nonhypertensive
59	Deficiency and other anemia
58	Other nutritional; endocrine; and metabolic disorders
117	Other circulatory disease
105	Conduction disorders
155	Other gastrointestinal disorders
663	Screening and history of mental health and substance abuse codes
257	Other aftercare
259	Residual codes; unclassified
96	Heart valve disorders
253	Allergic reactions
211	Other connective tissue disease

diagnosis are not ordered by attribute column (there is no fixed attribute columns). For this reason, we represented each visit with a set of diagnoses instead of a set of fixed attributes. However, as you can note in Table 23, to simplify the reporting, we assume the domain of each diagnosis is in $\{0, 1\}$, where 1 means that the patient is diagnosed with that particular diagnosis at the current visit, and 0 means that the patient is not diagnosed with that specific diagnosis at the current visit.

We first grouped our dataset by procedures that are represented here by the first primary procedure attribute $PRCCS1$. We also grouped the visits by patient ID $VisitLink$ and ordered each patient's visits by the $DaysToEvent$ attribute. We then built pairs of visits for the meta-action applied, and extracted the action terms from the pairs. Finally, we computed the likelihood and term confidence for each action term. As you can see from Table 23, the likelihood and term confidence of the action terms extracted are very high. The higher the likelihood of the action term, the more important the diagnoses and the more likely the diagnoses involved are the main reason of the procedure. In addition, the higher the term confidence, the more stable the meta-action and procedure result.

6.2.2.2 Action Rules Extraction and Evaluation

We extracted action rules using the association action rules extraction method. Commonly, we extract action rules from all the dataset; however, for the sake of this chapter, we extracted action rules from a subset of the dataset that contains patient visit records involving the meta-actions reported in Table 23. It is important to note that the meta-action attribute, $PRCCS1$, is not involved in the action rules extraction process. In addition, neither the patient ID, $VisitLink$, nor the ordering attributes, $DaysToEvent$, are used in the action rules extraction process.

Following this setting, we extracted the following two rules with their respective support and confidence, where patients stay alive; more precisely, the decision feature $DIED$ stays at 0:

$r_1 \;=\; (58, 1 \rightarrow 0) \wedge (59, 1 \rightarrow 0) \wedge (55, 1 \rightarrow 0) \Rightarrow (DIED, 0 \rightarrow 0)$

where $sup \;=\; 334$, and $conf \;= 85\%$

$r_2 \;=\; (62, 1 \rightarrow 0) \wedge (55, 1 \rightarrow 0) \wedge (106, 1 \rightarrow 0) \Rightarrow (DIED, 0 \rightarrow 0)$

where $sup \;=\; 101$, and $conf \;=\; 78\%$

Now, we would like to apply meta-actions to trigger those rules. Therefore, we need to pick up the action terms that cover the antecedent side of each rule and their corresponding meta-actions. If we apply the set of meta-actions $M_{\{48\}} = \{48\}$ to action rule r_1, we will get the following global confidence and execution confidence: $GlobConf(r_1) = 80\%$ and $ExConf(M_{\{48\}}(r_1)) = 68\%$; whereas, if we apply the meta-actions $M_{\{44\}} = \{44\}$ we will get : $GlobConf(r_1) = 100\%$ and $ExConf(M_{\{48\}}(r_1))$ $= 85\%$. In other words, if the patient has the following diagnoses $\{55, 59, 58\}$ = {Fluid and electrolyte disorders, Deficiency and other anemia, Other nutritional; endocrine; and metabolic disorders} it is better to perform a Coronary artery bypass graft (CABG), CCS code of 44, rather than Insertion; revision, replacement; removal of cardiac pacemaker or cardioverter/ defibrillator, CCS code of 48. However, depending on the cost of the meta-action and the satisfaction rate, a practitioner may make a different decision.

Similarly, we can apply meta-actions $M_{\{44\}} = \{44\}$, Coronary artery bypass graft (CABG), to trigger r_2 and resolve the diagnoses $\{62, 55, 106\}$ = {Coagulation and hemorrhagic disorders, Fluid and electrolyte disorders, Cardiac dysrhythmias} and we will get: $GlobConf(r_1) = 94\%$ and $ExConf(M_{\{48\}}(r_1)) = 73.32\%$.

We lack the cost of meta-actions in our dataset; hence, we cannot compute the *SatRate*. Nonetheless, this information can be obtained via consultation with a practitioner. If we assume that the cost of any given meta-action is the same and that λ is selected as a constant for each meta-action, then the *SatRate* will be equal to the *ExConf* and practitioners can base their decision on the best execution confidence.

Table 23: Influence matrix like table reporting support and confidence.

Meta-action	**Action terms with CCS codes**	**Like**	**TermConf %**
43	$(101, 1 \to 0)$	167	84
	$(55, 1 \to 0)$	165	91
	$(62, 1 \to 0)$	146	96
	$(106, 1 \to 0)$	135	88
	$(99, 1 \to 0)$	66	96
	$(55, 1 \to 0) \wedge (62, 1 \to 0)$	46	90
	$(55, 1 \to 0) \wedge (106, 1 \to 0)$	44	90
	$(158, 1 \to 0) \wedge (106, 1 \to 0) \wedge (99, 1 \to 0)$	14	87.5
	$(55, 1 \to 0) \wedge (158, 1 \to 0) \wedge (101, 1 \to 0) \wedge (99, 1 \to 0)$	8	100
44	$(101, 1 \to 0)$	181	85
	$(108, 1 \to 0)$	92	91
	$(62, 1 \to 0)$	97	98
	$(114, 1 \to 0)$	86	93
	$(58, 1 \to 0)$	193	91
	$(108, 1 \to 0) \wedge (106, 1 \to 0)$	32	94
	$(55, 1 \to 0) \wedge (106, 1 \to 0)$	42	91
	$(62, 1 \to 0) \wedge (55, 1 \to 0) \wedge (106, 1 \to 0)$	16	94
	$(55, 1 \to 0) \wedge (108, 1 \to 0) \wedge (58, 1 \to 0) \wedge (59, 1 \to 0)$	8	100

Table 23: (Continued)

Meta-action	**Action terms with CCS codes**	**Like**	**TermConf %**
45	$(101, 1 \rightarrow 0)$	262	78
	$(117, 1 \rightarrow 0)$	97	86
	$(105, 1 \rightarrow 0)$	85	83
	$(155, 1 \rightarrow 0)$	81	88
	$(663, 1 \rightarrow 0)$	201	85
	$(55, 1 \rightarrow 0) \wedge (59, 1 \rightarrow 0)$	33	89
	$(58, 1 \rightarrow 0) \wedge (257, 1 \rightarrow 0)$	40	77
	$(259, 1 \rightarrow 0) \wedge (663, 1 \rightarrow 0) \wedge (101, 1 \rightarrow 0)$	26	76
	$(58, 1 \rightarrow 0) \wedge (259, 1 \rightarrow 0) \wedge (663, 1 \rightarrow 0) \wedge (101, 1 \rightarrow 0)$	6	67
47	$(101, 1 \rightarrow 0)$	347	81
	$(117, 1 \rightarrow 0)$	135	88
	$(105, 1 \rightarrow 0)$	135	82
	$(96, 1 \rightarrow 0)$	99	93
	$(155, 1 \rightarrow 0)$	97	94
	$(117, 1 \rightarrow 0) \wedge (257, 1 \rightarrow 0)$	45	80
	$(259, 1 \rightarrow 0) \wedge (253, 1 \rightarrow 0)$	41	90
	$(117, 1 \rightarrow 0) \wedge (257, 1 \rightarrow 0) \wedge (101, 1 \rightarrow 0)$	24	83
	$(117, 1 \rightarrow 0) \wedge (105, 1 \rightarrow 0) \wedge (257, 1 \rightarrow 0) \wedge (101, 1 \rightarrow 0)$	9	75

Table 23: (Continued)

Meta-action	**Action terms with CCS codes**	**Like**	**TermConf %**
48	$(257, 1 \rightarrow 0)$	335	87
	$(96, 1 \rightarrow 0)$	152	90
	$(211, 1 \rightarrow 0)$	115	86
	$(106, 1 \rightarrow 0)$	147	94
	$(155, 1 \rightarrow 0)$	137	91
	$(59, 1 \rightarrow 0) \wedge (58, 1 \rightarrow 0)$	69	86
	$(55, 1 \rightarrow 0) \wedge (58, 1 \rightarrow 0)$	69	87
	$(58, 1 \rightarrow 0) \wedge (59, 1 \rightarrow 0) \wedge (55, 1 \rightarrow 0)$	27	80
	$(55, 1 \rightarrow 0) \wedge (58, 1 \rightarrow 0) \wedge (158, 1 \rightarrow 0) \wedge (99, 1 \rightarrow 0)$	11	100

Nowadays, action rules are used in several industries, and the healthcare industry among others is a very sensitive area. Results from action rules extraction process have to be thoroughly evaluated and analyzed to be used in such industries. Action rules are commonly constructed from feature values patterns and not from transition patterns. In this chapter, we used meta-actions to evaluate action rules and we introduced new evaluation metrics. We used the 2010 Florida State Inpatient Databases (SID), and extracted meta-actions and action rules from this dataset. We evaluated meta-actions applied to action rules with the different metrics and compared the results to traditional metrics.

6.3 Chapter Conclusion

In This chapter, we demonstrated how meta-actions could be used in healthcare systems. We introduced an action rules reduction mechanism. This mechanism allows the full execution of action rules while minimizing the number of side effects. This mechanism could be applied in the case where two patients react differently to a treatment recommended by an action rule. In fact, some patients diagnoses might be treated by the treatment (meta-action) applied because the meta-action used covers all their preconditions; other patients will not see improvements in their diagnoses because the action rule recommendation could not be fully executed. In addition, a worst situation where the patients might develop severe negative side effects could be avoided by the use of meta-actions to substitute action rules. We also introduced action rules evaluation metrics based on meta-actions and performed experiments that show the effectiveness of our metrics in comparison with the classical ones.

CHAPTER 7: CONCLUSION AND DISCUSSION

The analysis of treatments and side effects patterns in healthcare is a challenging task, especially when the applications of such knowledge have direct impact on patients' health. In this work, we identified and defined the meta-actions responsible in executing the action rules. We further studied the challenges introduced with applying meta-actions in healthcare systems. We used the tinnitus handicap dataset and the Healthcare Cost and Utilization Project (HCUP) Florida State Inpatient Databases (SID 2010) to validate and evaluate our approaches.

We studied meta-actions and their effects that represent medical treatments. We defined and considered two representations of meta-actions and mined their effects. The first representation of meta-action effects was based on action terms and the second was based on action sets. Furthermore, we defined evaluation metrics for both representations and evaluated our mining methods. The results of the meta-actions mining for both representation techniques and both provided datasets show that the effects of medical treatments are stable with high confidence with regards to patients' preconditions.

The analysis of mined negative side effects shows that their confidence with regards to patients' preconditions in term of diagnoses is very low, which means that side-effects do not depend on patients' diagnoses. They rather depend on the patient's state with regards to co-morbid conditions and stable features. Their prediction also depends on the patients' overall state based on the cluster analysis. When treatment (meta-actions) and negative side effects are known, we proposed three patients grouping schemes for personalized action rule extraction, and noted that the meta-actions based grouping returned the best outcome in terms of confidence and support.

We used the meta-actions effects to present alternative substitute action rules when their original successful execution is improbable and they result in severe side effects. We showed that the new reduced action rules have higher execution confidence when coupled with meta-actions. In addition, we presented action rules evaluation based on meta-actions and compared both concepts. We introduced new evaluation metrics for action rules execution and compared them to traditional metrics. Our evaluation metrics showed more accurate estimations for action rules. Moreover, we presented an attempt to create a personalized treatment recommendation system for patients.

The main shortcomings and challenges that we encountered in this work are the unavailability of detailed ontology for diagnoses and their order of severity. This makes treatment recommendations flawed since doctors may decide to start by prescribing treatments for urgent and severe problems when faced with a multi-diagnoses pathological state. Furthermore, The datasets that we received are highly incomplete and provide inconsistent structures.

REFERENCES

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 487–499, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers Inc.

[2] J. Bailey, A. Poulovassilis, and P. T. Wood. Analysis and optimisation of event-condition-action rules on xml, 2001.

[3] H. Cost and U. P. (HCUP). Clinical classifications software (ccs), http://www.hcup-us.ahrq.gov.

[4] H. Cost, U. P. (HCUP), A. for Healthcare Research, and Quality. Clinical classifications software (ccs) for icd-9-cm, http://www.hcup-us.ahrq.gov/toolssoftware/ccs/ccs.jsp.

[5] M. d. D. Felicia C Allen-Ramey, Shaloo Gupta. Patient characteristics, treatment patterns, and health outcomes among copd phenotypes, 2012.

[6] D. Fensel. *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*. Springer, Syracuse, NJ, USA, 2003.

[7] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry. Using collaborative filtering to weave an information tapestry. *Communications of ACM*, 35(12):61–70, 1992.

[8] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. Eigentaste: a constant time collaborative filtering algorithm. *Information Retrieval*, 4(2):133–151, 2001.

[9] N. Goodman. Statistical analysis based on a certain multivariate complex gaussian distribution (an introduction). *Annals of mathematical statistics*, pages 152–177, 1963.

[10] S. Greco, B. Matarazzo, N. Pappalardo, and R. Slowinski. Measuring expected effects of interventions based on decision rules. In *Journal of Experimental and TheoreticalAI, Taylor and Francis*, 17, pages 103–118. Taylor and Francis, 2005.

[11] J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Applied statistics*, pages 100–108, 1979.

[12] Z. He, X. Xu, S. Deng, and R. Ma. Mining action rules from scratch. *Expert Syst. Appl.*, 29(3):691–699, Oct. 2005.

[13] J. Hipp, U. Güntzer, and G. Nakhaeizadeh. Algorithms for association rule mining: a general survey and comparison. *SIGKDD Explor. Newsl.*, 2(1):58–64, June 2000.

[14] A. H. inc. Audax health, http://www.audaxhealth.com/.

[15] F. H. inc. Flatiron, oncologycloud, http://flatiron.com/.

[16] N. Jiang and L. Gruenwald. Research issues in data stream association rule mining. *SIGMOD Rec.*, 35(1):14–19, Mar. 2006.

[17] K. S. Kolibaba, J. A. Sterchele, A. D. Joshi, M. Forsyth, E. Alwon, H. Beygi, , and G. T. Kennealey. Demographics, treatment patterns, safety, and real-world effectiveness in patients aged 70 years and over with chronic lymphocytic leukemia receiving bendamustine with or without rituximab: a retrospective study. *Therapeutic Advances in Hematology*, 4(3):157171, 2013.

[18] P. Lorigan, M. Marples, M. Harries, J. Wagstaff, A. Dalgleish, R. Osborne, A. Maraveyas, S. Nicholson, N. Davidson, Q. Wang, L. Pericleous, U. Bapat, and M. Middleton. Treatment patterns, outcomes, and resource utilization of patients with metastatic melanoma in the u.k. *British Journal of Dermatology*, page 8795, 2014.

[19] U. D. of Health & Human Services. Summary of the hipaa privacy rule, http://www.hhs.gov/ocr/privacy/hipaa/understanding/summary/.

[20] Z. Pawlak. Information systems - theoretical foundations. *Information Systems Journal*, 6:205–218, 1981.

[21] Y. Qiao, K. Zhong, H. Wang, and X. Li. Developing event-condition-action rules in real-time active database. In *Proceedings of the 2007 ACM symposium on Applied computing*, SAC '07, pages 511–516, New York, NY, USA, 2007. ACM.

[22] Z. Raś and A. Dardzińska. Action rules discovery based on tree classifiers and meta-actions. In *Proceedings of the 18th International Symposium on Foundations of Intelligent Systems*, ISMIS '09, pages 66–75. Springer, 2009.

[23] Z. Raś and A. Dardzinska. From data to classification rules and actions. *Int. J. Intell. Syst.*, 26(6):572–590, June 2011.

[24] Z. Raś, A. Dardzinska, L. Tsay, and H. Wasyluk. Association action rules. In *Data Mining Workshops. ICDMW '08. IEEE International Conference*, pages 283–290, 2008.

[25] Z. Raś and A. Wieczorkowska. Action-rules: How to increase profit of a company. In *Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery*, PKDD '00, pages 587–592, London, UK, 2000. Springer.

[26] Z. Raś, E. Wyrzykowska, and H. Wasyluk. Aras: action rules discovery based on agglomerative strategy. In *Proceedings of the 3rd ECML/PKDD international conference on Mining complex data*, MCD'07, pages 196–208, Berlin, Heidelberg, 2008. Springer-Verlag.

[27] J. Rauch. Action4ft-miner module, lisp-miner project.

[28] J. Rauch and M. Šimůnek. Action rules and the guha method: Preliminary considerations and results. In *Proceedings of the 18th International Symposium on Foundations of Intelligent Systems*, ISMIS '09, pages 76–87. Springer-Verlag, 2009.

[29] P. Resnick and H. R. Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.

[30] F. Tao, F. Murtagh, and M. Farid. Weighted association rule mining using weighted support and significance framework. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '03, pages 661–666, New York, NY, USA, 2003. ACM.

[31] M. Thangamani, P. Thangaraj, and Bannari. Automatic medical disease treatment system using datamining. In *Information Communication and Embedded Systems (ICICES), 2013 International Conference on*, pages 120–125, Feb 2013.

[32] H. Touati, J. Kuang, A. Hajja, and Z. Ras. Personalized action rules for side effects object grouping. *the Special Issue on "Knowledge Discovery", G. Wang (Ed.) , International Journal of Intelligence Science (IJIS)*, 3(1A):24–33, 2013.

[33] H. Touati and Z. Ras. Mining meta-actions for action rules reduction. *Fundamenta Informaticae*, 127(1-4):225–240, 2013.

[34] H. Touati, Z. Ras, J. Studnicki, and A. Wieczorkowska. Mining surgical meta-actions effects with variable diagnoses number. In *Proceedings of ISMIS 2014 in Roskilde*, Denmark, 2014. Springer.

[35] A. Tuzhilin, G. Adomavicius, G. Adomavicius, and A. Tuz. Discovery of actionable patterns in databases: The action hierarchy approach. In *In KDD*, pages 111–114. AAAI Press, 1997.

[36] A. Tzacheva and Z. Raś. Association action rules and action paths triggered by meta-actions. In *Proceedings of the 2010 IEEE International Conference on Granular Computing*, pages 772–776. IEEE Computer Society, 2010.

[37] K. Wang, Y. Jiang, and A. Tuzhilin. Mining actionable patterns by role models. In *Data Engineering, ICDE '06. Proceedings of the 22nd International Conference*, pages 16–26, 2006.

[38] H. Wasyluk, Z. Raś, and E. Wyrzykowska. Application of action rules to hepar clinical decision support system. *Journal of Experimental and Clinical Hepatology.*, 4(2):46–48, 2008.

[39] H. Zhang, Y. Zhao, L. Cao, and C. Zhang. Combined association rule mining. In *Proceedings of the 12th Pacific-Asia conference on Advances in knowledge discovery and data mining*, PAKDD'08, pages 1069–1074, Berlin, Heidelberg, 2008. Springer-Verlag.

[40] X. Zhang, Z. Raś, P. Jastreboff, and P. Thompson. From tinnitus data to action rules and tinnitus treatment. In *Granular Computing (GrC), 2010 IEEE International Conference on*, pages 620–625, Aug.