Cloud services selection based on rough set theory

Yongwen Liu

Thèse
de doctorat
de l'UTT

Yongwen LIU

Cloud Services Selection Based on Rough Set Theory

Spécialité :
Ingénierie Sociotechnique des Connaissances, des Réseaux et du Développement Durable

2016TROY0018

Année 2016

utt
université de technologie
Troyes

THESE

pour l'obtention du grade de

DOCTEUR de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

Spécialité : INGENIERIE SOCIOTECHNIQUE DES CONNAISSANCES, DES RESEAUX ET DU DEVELOPPEMENT DURABLE

présentée et soutenue par

Yongwen LIU

le 17 juin 2016

Cloud Service Selection based on Rough Set Theory

JURY

M. H. SNOUSSI	PROFESSEUR DES UNIVERSITES	Président
M. A. AHMED	ASSISTANT PROFESSOR	Examinateur
M. M. ESSEGHIR	MAITRE DE CONFERENCES	Directeur de thèse
M. M. Y. GHAMRI-DOUDANE	PROFESSEUR DES UNIVERSITES	Rapporteur
Mme L. MERGHEM-BOULAHIA	MAITRE DE CONFERENCES - HDR	Directrice de thèse
M. S.-M. SENOUCI	PROFESSEUR DES UNIVERSITES	Rapporteur

ABSTRACT

This thesis presents an application of rough set theory in cloud services selection. The main purpose of doing this is to apply a theory to real life to guide our practice action. We implement lots of tests on huge amount of dataset and the experimental results verified the efficiency of our proposal. With the development of cloud computing technique, users enjoy various benefits that high technology services bring. However, with the technique maturity, there are more and more cloud service programs emerging. So it is important for users to choose the right cloud service. For cloud service providers, it is important to make a progress for the cloud services they provided, thus to win more customers and expand the scale of the cloud services.

rough set theory is a good data processing tool to deal with uncertain information. In this work, we propose a method using the rough set theory in cloud service selection and an example to illustrate the practice and analyze the feasibility of it. The main contributions of this work are: First, we perform the program experiments with large scale dataset to verify the feasibility and practicality. The performance results with a large scale of dataset can help cloud services users to make the right decision and help cloud services providers to target their improvement about the cloud services programs; Second, We proposed the cloud services selection approach to evaluate parameters importance based on the users preferences using rough set theory.

The performance of program code is by Java language. They are executed sequentially on a processor Intel Core2 Duo CPUs x64. The total main memory is 8 Gigabyte and the operating system is Windows 8. Results collected during the experiments on a number of small datasets and lots of huge datasets for selecting a classified attributes show that the proposed application is an efficient approach with good practical value.

Keywords: Cloud computing; Rough Sets; Decision making; Decision support systems; Classification; Web services;

ACKNOWLEDGEMENTS

First of all, I would like to express my special appreciation and thanks to my supervisors Moez ESSGHIR and Leila MERGHEM BOULAHIA, they are tremendous mentors for me. They support continuously my Ph.D study and related research with their motivation, patience, sense of confidence on me and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis. I carried out my work in ERA (Environnement des Reseaux Autonomes) Team at Universite de Technologie de Troyes. I would like to thank my lab mates for the discussions.

I would like to thank the rest of my thesis committee: Prof. Sidi-Mohammed Senouci, Prof. Yacine Ghamri-Doudane, Prof. Snoussi Hichem and Assistant Prof. Ahmed Atiq, for their insightful comments and encouragement, which incentes me to widen my research from various perspectives.

I would like to thank China Scholarship Council that provides fund to complete my study.

I would like to thank all of my friends who supported me to strive towards my goal.

I would like to thank my family for supporting me spiritually throughout writing this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this section, firstly, we state the research background of our study. Secondly, we present the problems to need solve in cloud service selection. Thirdly, we introduce the objectives, scope, contributions and structure of our study. Lastly, we summary this chapter.

1.1 Background

Cloud computing as a new information technique has been developing rapidly in recent years, which raises the tide for the whole information community. It offers many potential benefits to companies or organizations by making information technology services available as a commodity. When companies or organizations contract cloud services, such as software application, data storage, and data processing capabilities, it can improve their efficiency and ability of operation. Cloud computing as a tool for helping cloud services users provide reliable, innovative and timely services.

Since cloud services can reduce the cost and complexity of owning and operating computers and networks, they are popular. Cloud service users do not have to invest in information technology infrastructure, maintenance equipment, purchase and upgrade hardware or software, the benefits are low up-front costs, high returns in future, rapid deployment, customization, flexible use, and solutions that can allow the organizations to free up resources to focus on innovation and product development. In addition, cloud service providers that have specialized in a particular area can bring advanced services that some company themselves might not be able to afford or develop in short time. However, challenges are always there for us to surpass. Like any new technology, the adoption of cloud computing is not free from issues. Some of the most important challenges are as follows.

1. Security and privacy

The security and privacy are the main challenge to cloud computing, because it concerns of businesses thinking of adopting it. As the valuable enterprise or institution data outside their corporate firewall, it will concern some issues, such as access control, identify and rights management, privacy and integrity, verification and certification etc. Specially, we should prevent from hacking and various attacks to cloud infrastructure, even if only one site is attacked, it would affect multiple users.

2. Delivery and billing

Budgeting and assessment of the costs involved are difficult due to the on-demand nature of the services, although where possible, the providers have some good comparable benchmarks to offer. Some times, the service-level agreements(SLAs) of the providers are not adequate to guarantee the availability and scalability. If there is no a strong service quality guarantee, the enterprises or institutions won't want to move their businesses to cloud.

3. Interoperability and Portability

The cloud computing interoperability categories to consider are platform interoperability, management interoperability, publication and acquisition interoperability. The main kinds of cloud computing portability to consider are data portability, application portability, and platform portability. Users should have the leverage of migrating in and out of the cloud and switching providers whenever they want, and there should be no lock-in period. Cloud computing services should have the capability to integrate smoothly with the on-premise IT.

4. Reliability and Availability

As the adoption of cloud computing becomes widespread, and users demand 24/7 access to their services and data, availability and reliability remains a challenge for cloud service providers everywhere. Failures are inevitable in complex systems. Cloud providers still lack round-the-clock service; this results in frequent outages. Cloud service providers should consider in relation to their cloud services at four main categories: 1) maximize service availability to users, 2) Minimize the impact of any failure on users, 3) maximize service performance, 4) maximize business continuity.

5. Performance and Bandwidth Cost

Network performance and bandwidth are critical to cloud success. Enterprises can save money on hardware but they have to spend more for the bandwidth. This can be a low cost for smaller applications but can be significantly high for the data-intensive applications. Delivering intensive and complex data over the network requires sufficient bandwidth. Because of this, many enterprises are balancing the cost before switching to the cloud.

For above challenges in cloud computing, the researchers have done a lot of works, and some woks in the continuous. In literatures [1] $\sim$ [9], the authors state and analyze all the kinds of security issues that not only threat cloud users but also cloud providers, even threat the construction of the IT infrastructure. The researchers who study in concert with cloud security fields give some corresponding solutions[10]. Literature [11] proposes introducing a Trusted Third Party which is responsible for ensuring specific security characteristics within a cloud environment. Users of adopting the cloud services fear their sensitive data leakage and loss in a way. For this problem, Miranda and Sinani [12] proposed a client-based privacy manager for cloud computing to help users reduce data security risk, additionally, that provides privacy-related benefits. In addition to this, the researchers do a lot of works about all kinds of security issues in adapting the cloud computing technology. As an important service of cloud computing, cloud storage allows users move their data from their local storage system to the cloud. Cloud users do not have to care the complexity of hardware and software managements and deployments. It offers great convenience to users, it brings a number of security issues towards the data information[13]. Literature [14], [15] and [16] proposed different secure cloud auditing protocols and privacy-preserving auditing mechanisms through the third party.

Some fundamental challenges for wide adoption of cloud computing are presented in literature[17], such as service life cycle optimization, scalable and dependable service platforms and architectures and adaptive self-preservation. The solution in this work focuses on a holistic approach to cloud service provisioning and discuss that a a single abstraction for multiple coexisting cloud architectures is imperative for a broader cloud service ecosystem. The authors assumed that clouds are available as private and public, they design a toolkit which the toolkit aims to provide a foundation for a reliable, sustainable, and trustful cloud computing industry, and optimizing the whole service life cycle in it.

Cloud services can realize benefits for cloud users. As a commercial operation model, more and more cloud service providers emerge, cloud users need to choose the appropriate cloud providers, that is the shop around. However, it is a sophisticated task to do

this for an enterprise or an organization. Our work focus on helping the cloud service users make a decision to choose the right providers.

Before we buy a product, we first know its applications, performance and effectiveness, then we shop around in different providers, finally, we make a decision. It looks a simple process. However, when we buy a service, it becomes complex. Users have to make sure what services are they needed, they how to compare the providers, how to assess the providers and their services. As we mentioned above, we devote our efforts to help cloud users to select appropriate providers. At the same time, we also dedicate that providers improve the quality of products to have more advantages in competition.

1.2 Problems statement and Solutions

The decision making process is not easy, no matter we buying a house, moving across the country, quitting a job, or just deciding what film to see, can all drain our willpower. For some companies or institutions, it is very important to make a right decision because it concerns their future development. For example, cloud services are vital part of today's society - many of companies or institutions want to or already move their data into the cloud. All this complexity is hidden from the cloud user, and the global nature of the market providers keen competition. Costs for cloud-based services are, by and large, cheap, and in some cases the services are free at the point of use. Data may be stored under foreign legal jurisdictions, potentially allowing governments or other organisations access to certain aspects of users' operations, thus, it might cause the confidential information divulged. So cloud users choosing the services to decide what level of information assurance their data requirements.

About our work, we aim to assess the cloud services or providers to help cloud users make a decision for choosing the right services. It is very difficult to develop a comprehensive assessment of cloud service providers without some structure or framework. So, the problems we need solve are 1) how to establish a framework for extracting useful information to help cloud users make a right decision; 2) how to evaluating the parameters importance of cloud services selection. For solutions, firstly, we need choose the appropriate data mining techniques to support our study. Some of the most common data mining techniques or algorithms in use today are neighbor relationship, clustering, decision trees, neural networks and so on. Each mining algorithm or technique fits into the different scope of application and its characteristics. In our study, we choose rough set theory as the research tool. In the latter chapter, we will provide the reason why we choose it. After, we give a framework to assess the cloud service providers using rough

set theory, then we provide an approach for evaluating the importance of parameters and ranking them in cloud services selection.

1.3 Objectives and Scope

This research takes up the following objectives:

a. To develop a framework for cloud services selection using rough set theory based on discernibility matrix to extract rules to help cloud users make a decision.

b. To assess importance of cloud services parameters and rank them using rough set theory

c. To do a comparison between the proposed technique with the related works.

The scope of this research falls within data classification, decision making using rough set theory.

1.4 Contributions

The specific contributions of this thesis correspond to the free factors as describe earlier, which are

a. Provide a process for obtaining the rules to help cloud users decision making using rough set theory

b. Reduce redundancy parameters for assessing cloud services

1.5 Structure of this thesis

This is an outline of the thesis. This gives a summary of each chapter of the thesis.

Chapter 1: Introduction

The aim of this chapter is to introduce our topic. In this chapter, we are discussing the relevant concepts related to our topic like cloud computing technique, cloud service models, customer requirements. Also, the need for our study is introduced in order to what is it focused on and what are the problems we need solve in our study. Here, we give our research questions and purpose as the clear road map of our study. We are interested in using rough set theory to establish the framework for help cloud users to

choose the cloud services or cloud services providers. We are also interested in using rough set theory to assess the importance of parameters of cloud services and rank them. Like this, it guides the users to make a right decision from different cloud service providers.

Chapter 2: The cloud service selection technique

In this chapter we introduce some basic concepts such as cloud computing, service composition. The purpose of this chapter is to present and discuss already exist classification techniques and algorithms. We are carrying a quantitative study and our research design in order to make our study objective. In fact, we are not interested in comparing the benefits or disadvantages of all the classification techniques and algorithms but rather trying to answer we choose rough set theory as the research tool to solve the questions. In this regard, we make a description for the classification techniques and algorithms and give the reason we choose rough set approach to carry out our study. Various challenges of cloud service selection are presented in this chapter. Researchers done a lot of related works and obtain some achievement. We present existing related works and summarize them that researchers proposed and some limits for their application.

Chapter 3: Related Knowledge

In this chapter, we present all the related knowledge that are important to our study. Concepts such are knowledge space, lower and upper approximations, indiscernibility relations, attributes reduction and extract rules are discussed. Also, we give an instance to intensively understand these concepts. We try to enhance our main theories involved in our study and to answer our research question.

Chapter 4: Application of the rough set theory in cloud services selection

This chapter discusses the application in cloud service selection using rough set theory. We briefly introduce the related works, and summarize some already exist research approaches for this part. The main work in this chapter is to discuss the details we carried out using rough set theory. We also compare our works with others.

Chapter 5: Evaluation of parameters importance in cloud service selection using rough set theory

In this chapter, we discuss how to evaluate the parameters importance in cloud service selection. A general description of the approach we proposed was done for computing the weight of parameters and ranking them using rough set theory. We implement the experiment. Result analysis was done in order to verify the validation of the approach we proposed.

Chapter 6: Conclusions and future works

In this chapter, we have a conclusion for our study, such as the solutions for cloud service selection problems. For the future works, we analyze the cloud computing develepment trends and the mains problems currently, we provide the research work next step.

1.6 Conclusion

In this section, we presented the background, purpose and problems about our study. We listed the structure and major context of every chapter. In the next chapter, we will state the cloud service selection technique currently.

Chapter 2

The cloud service selection technique

2.1 Introduction

Cloud Computing is an emerging computing paradigm. It shares massively scalable, elastic resources (e.g., data, calculations, and services) transparently among the users over a massive network[19]. More and more resources are encapsulated as services and form a cloud market, which brings up numerous research challenges. The area of cloud service selection is one of the key challenges. Cloud services as special commodities, company should be able to buy their service requirements from primary cloud service providers or cloud brokers who manages the accounts of hundreds or thousands of clients.

First, how to select the best service out of the huge resources pool for consumers; how to manage effectively the cloud clients and help them chose the appropriate services for the broker. To solve these problems, researchers have designed uniform cloud market platforms for publishing services and locating services for service providers and users where all suppliers compete on price for similar services. Additionally, researchers proposed assistant approaches for choosing appropriate services based on decision-making techniques such as rough set, neural network and so on. In second, with the tough competition between cloud service providers, it becomes difficult for service providers supplying simple service selection or service composition, which is considered an NP-hard problem[47]. To solve related service composition problems, researchers have done a lot of work also.

This chapter is structured as follows. In section 2.2, we begin by describing the definition of cloud computing, we then give the deployment models and service models

of cloud computing. In section 2.3, we introduce the related works about cloud service selection, which includes the challenges and existing the solutions etc. The techniques of ranking and recommend system of cloud service selection will be introduced in this section.

2.2 Cloud computing

The NIST (National Institute of Standards and Technology) defines cloud computing as follows:

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [21].

Cloud computing as a service model for computing service with the character "pay as you go" similar to the utility model (gas, telecommunication, electricity and water), once cloud users are connected to computing cloud, they can consume as much service as they would like, and they pay for the resources consumed [22]. Resources such as storage, network, computing platform and solution stacks are provisioned as services. The resource utilization and operational efficiency can be higher across a shared computing resources pool. The price of the service to cloud users may well be lower from a cloud provider compare with deploying applications and possibly configuration settings for the application-hosting environment.

With the mature of cloud computing, cloud users can have on demand self-service for computing capabilities in different platforms, such as server time and network storage when needed, through a cloud services provider. When cloud users hope to move their business to cloud computing platform, they should evaluate the different technologies and configurations and determine the specific parts of the cloud computing scope that meet their needs. The factors to be considered such as deployment models, service models and economic considerations.

Deployment Models. Depending on the kind of cloud deployment, the cloud may have limited private computing resources, or may have access to large quantities of remotely accessed resources. The following deployment models present a number of trade-off in how customers can control their resources, and the scale, cost, and availability of resources.

- Private cloud[42]

  The cloud infrastructure is operated solely for an organization be as a specific client. This model does not bring much for cloud users in terms of cost efficiency comparing to buying, building and managing users' own infrastructure. Still, it brings in tremendous value from a security point of view. Because many organizations adapting the cloud face challenges and have concerns related to data security, these concerns are taken care of by this model.

- Community cloud[42]

  In the community deployment model, the cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). This helps to further reduce costs comparing to a private cloud due to its sharing. This helps to further reduce costs as compared to a private cloud, as it is shared by larger group. For example, various state-level departments can utilize a community cloud to manage applications and data relating to local information related to infrastructure,such as hospitals, electrical stations, police stations, etc. It may be managed by the organizations or a third party and may exist on premise or off premise.

- Public cloud[42]

  The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services. In this deployment model, services and infrastructure are provided to various cloud users. This model is best suited for cloud users who do not want to invest largely in infrastructure whereas they can manage load spikes, host SaaS applications, utilize interim infrastructure for developing and testing applications, and manage application which they consumed. This deployment model helps to reduce capital expenditure and bring down operational IT costs.

- Hybrid cloud[42]

  The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but that are bound together by standardized or proprietary technology enabling data and application portability. In this deployment model, cloud users take advantage of cost benefits by keeping shared data and applications on the public cloud meanwhile they enjoy secured applications and data hosting on a private cloud[40] .

- On-site private cloud[23]

  The security perimeter for this deployment model extends around both the subscriber's on-site resources and the private cloud's resources. The private cloud may be centralized at a single subscriber site or may be distributed over several subscriber sites.The subscriber implements the security perimeter, which will not guarantee control over the private cloud's resources, but will enable the subscriber to exercise control over resources entrusted to the on-site private cloud.

Generally, cloud services models as new business models can be classified in three categories:

- Cloud Infrastructure as a service (IaaS): is the virtual delivery of computing resources in the form of hardware, networking, and storage services. The cloud users can deploy and run arbitrary software they needed. IaaS can also include the delivery of operating systems and virtualization technologies to manage its own virtual infrastructure resource which typically constructed by virtual machine hosted by the IaaS providers[24][18]. The goal of IaaS is to avoid buying and installing new resources while they can be easily rent.

- Cloud Platform as a service (PaaS): is an abstracted and integrated cloud-based computing environment that supports the development, running, and management of applications, in which applications are hosted by service providers and made available to customers over the Internet. PaaS focuses on providing the higher level capabilities more than just virtual machines required to supports applications[24]. In PaaS, operating system features can be changed and upgraded frequently.

- Cloud Software as a service (SaaS): is not a stand-alone environment. Instead, these applications and services are frequently used in combination with lots of other cloud and on premise models. Companies need their SaaS applications to couple with other applications and platforms on their own data center and with other cloud platforms. The service providers do all the upgrades and patching while keeping the infrastructure running.

Figure 2.1 visualizes the relationship between these deployment and service models.

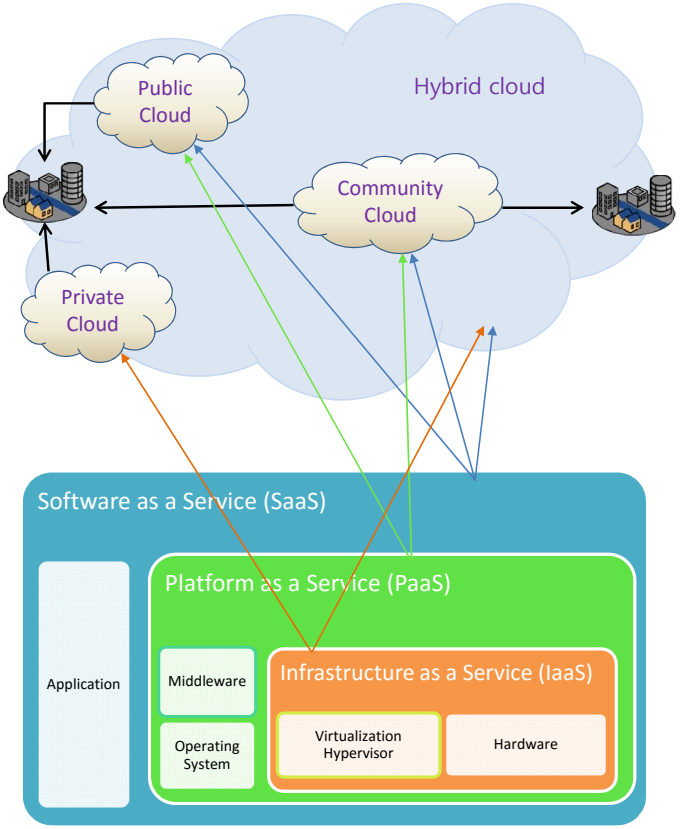


Figure 2.1: Cloud computing deployment and service models

2.3 Related techniques of cloud service selection

Lots of knowledge the making-decision needs for business and research are hidden in big data. Classification is a form of data analysis. It can extract model for describing important data set or predicting future trend of data. Classification is used to predict the categorical label of data objects.

On general, classification can be roughly divided into two types of traditional classification algorithms and base on soft computing method. They mainly include Similar functions, Association rule classification algorithm, K nearest neighbor classification algorithm, Decision tree classification algorithm, Bayesian classification algorithm based on fuzzy logic, Genetic algorithms, Rough sets and Neural network classification algorithm etc.

Each algorithm has different capabilities and characteristics to complete various tasks. A lot of classification algorithms are proposed by the researchers who working in machine learning, expert system, the statistics and neurobiology and so on. We usually evaluate the different classification algorithms by some indexes such as accuracy, speed,

robust, scalability, interpretation etc.

There are many classification and decision-making algorithms. We introduce some common approaches such as Decision tree, Bayes, Association Rule and SVM.

2.3.1 Decision tree classification algorithm

A decision tree is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility[25]. It is one way to display an algorithm.

Decision tree is commonly used in operations research, specifically in decision analysis, to help identify a strategy most likely to reach a goal. Decision tree analysis procedures can address some complexities of decisions with significant uncertainty, 1) there are a lot of different factors that must be taken into account when making a decision, 2) some specified decision alternative cannot be predicted with certainty, 3) consider the possibility of reducing the uncertainty in making decision by collecting additional information[25]. If in practice decisions have to be taken online with no recall under incomplete knowledge, a decision tree should be paralleled by a probability model as a best choice model or online selection model algorithm. Another use of decision tree is as a descriptive means for computing conditional probability.

To design decision tree classifier there can be three steps: 1) choosing the appropriate tree structure, 2) choosing the feature subsets to be used at each internal node, 3)choosing the decision rule or strategy to be used at each internal node. The main objectives of decision tree classifier are: 1) to classify correctly as much of the training sample as possible; 2) generalize beyond the training sample so that unseen samples could be classified with as high of an accuracy as possible; 3) be easy to update as more training sample becomes available (e.g., be incremental); 4) and have as simple a structure as possible.

The construction of decision tree classifier can roughly be divided into four categories: The top-down approach, the bottom-up approach, the tree growing-pruning approach and the hybrid approach. In a bottom-up way, a decision tree is constructed using the training set. It is using some distance measure, the two classes with the smaller distance are merged to form a new group. We compute the mean vector and the covariance matrix for each group from the training samples of classes, and this step is repeated until one is left with one group at the root. In this way to construct a tree, the more obvious discrimination is done first, and more subtle ones at later stages of the tree. In top-down approach to tree design, sets of classes can be successively

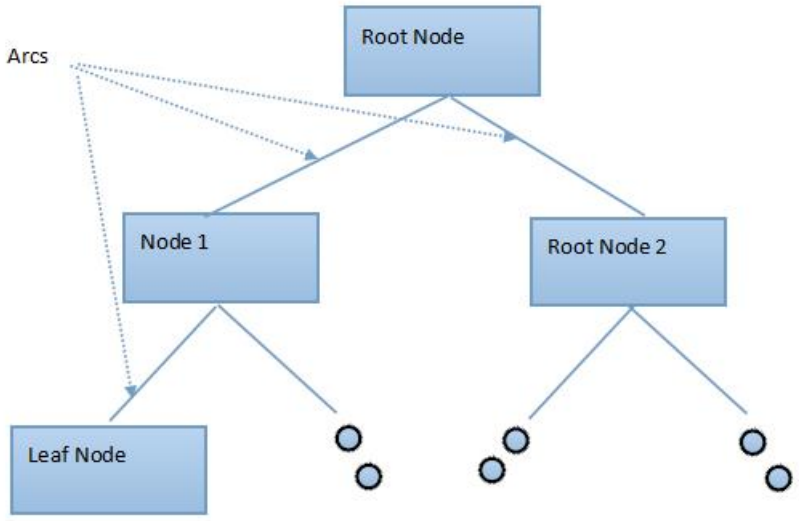


Figure 2.2: Basic decision tree structure

decomposed into smaller subsets of classes.

Decision tree classification algorithm also known as a greedy algorithm is heuristic, which can deduce the classification rules of decision tree representations from a set of disorder instances without rules. Decision tree classification algorithm is one of the most widely used classification algorithms, which is robust for noisy data and can learn the disjunctive normal form of a logic expression.

A decision tree consists of nodes and arcs which connect nodes. To make a decision, one starts at the root node, and asks questions to determine which are follow, until one reaches a leaf node and the decision is made. This basic structure is shown in Figure 2.2.

Each internal node of decision tree represents a test on an attribute (e.g. Whether a coin flip comes up heads or tails), each individual branch represents a test output and each leaf node represents class label or class distribution (decision taken after computing all attributes. The top-most node of tree is the root node. The paths from root to leaf represents classification rules. Decision tree algorithm classify the unknown sample by comparing the value of training samples and test dataset. The generation process as follows:

Firstly, according to the training data set to construct decision tree. In fact, building the decision tree model is the process of machine learning to obtain knowledge from data. The root node of decision tree as a start, using the classification attributes (for quantitative attributes, they should be discretized) classify the samples by choosing the corresponding test attributes recursively. Once an attribute appears on a node, it

cannot be emerge on any offspring of this node, test attribute is chosen according to certain heuristic information or statistic information (such as information gain). The second stage is tree pruning, tree pruning tries to detect and remove the noisy and the isolated points of training data set, and to eliminate the exception of model at the most of extent. The tree becomes more smaller with low complexity after pruning, and the classification is more faster and better for independent inspection data correctly.

ID3 (Iterative Dichotomisers) and C4.5 are earliest decision trees algorithms introduced by Ross Quinlan[26] for inducing classification models from a dataset. ID3 is the precursor to the C4.5 algorithm, and C4.5 is an extension of earlier ID3 algorithm. They are often referred to as statistical classifiers. They are effective for small-scale training samples. For large-scale dataset, its very complex to structure their decision tree and the classification efficiency is not high. To solve the shortages of the algorithms, there are some improved decision tree algorithms, such as a fuzzy decision tree algorithm based on C4.5 [27], an improved ID3 decision tree algorithm [28], they improve the classification accuracy and ability of induction.

The advantages of decision tree classifier[26]:

1)It can assign specific values to problem, decisions, and outcomes of each decision. This reduces ambiguity in decision-making. Every possible scenario from a decision finds representation by a clear fork and node, enabling viewing all possible solutions clearly in a global view.

2)It allows for comprehensive analysis of the consequences of each possible decision, such as what the decision leads to, whether it ends in uncertainty or a definite conclusion, or whether it leads to new issues for which the process needs repetition. Moreover, it allows for partitioning data in a much deeper level, not as easily achieved with other decision-making classifiers such as logistic regression or support of vector machines.

3)It can be combined with other decision techniques. Sophisticated decision tree models are implemented for custom software application, which can use historic data to apply a statistical analysis and make predictions regarding the probability of events. For instance, the decision tree analysis helps to improve the decisions-making capability of commercial banks by assigning success and failure probability on application data to identify borrowers who do not meet the traditional, minimum-standard criteria set for borrowers, but who are statistically less likely to default than applicants who meet all minimum requirements.

4)In single stage classifiers, only one subset of features is used for discriminating among all classes. This feature subset is usually selected by a globally optimal criterion, such as maximum average inter-class separability. In decision tree classifiers,

on the other hand, one has the flexibility of choosing different subsets of features at different non-terminal nodes of the tree such that the feature subset chosen optimally discriminates among the classes in that node. This flexibility may actually provide performance improvement over a single-stage classifier.

5)It focuses on the relationship among various events and thereby, replicates the natural course of events, and as such, remains robust with little scope for errors, provided the data is correct.

The disadvantages of decision tree classifier:

1)The reliability of the information in the decision tree depends on feeding the precise internal and external information at the onset. Even a small change in input data can at times, cause large changes in the tree. Changing variables, excluding duplication information, or altering the sequence midway can lead to major changes and might possibly require redrawing the tree.

2)The decisions contained in the decision tree are based on expectations, and irrational expectations can lead to flaws and errors in the decision tree. Although the decision tree follows a natural course of events by tracing relationships between events, it may not be possible to plan for all contingencies that arise from a decision, and such oversights can lead to bad decisions.

3)Decision trees, while providing easy to view illustrations, can also be unwieldy. Even data that is perfectly divided into classes and uses only simple threshold tests may require a large decision tree. Large trees are not intelligible, and pose presentation difficulties.

4)There may be difficulties involved in designing an optimal decision tree classifier. The performance of a decision tree classifier strongly depends on how well the tree is designed.

5)For data including categorical variables with different number of levels, information gain in decision tree are biased in favor of those attributes with more levels.

2.3.2 Bayes classifier

Bayes classifier is based on applying Bayes theorem with independence assumptions between the features. This Classifier is named after Thomas Bayes (1702-1761)[29], who proposed the Bayes Theorem.

Bayesian classification provides practical learning algorithms and prior knowledge and observed data can be combined. Bayesian Classification provides a useful perspective for understanding and evaluating many learning algorithms[30]. It calculates

explicit probabilities for hypothesis and it is robust to noise in input data.

The main idea of Bayes classifier is that the role of a class to predict the values of features for members of that class. Examples are grouped in classes because they have common values for the features. Such classes are often called natural kinds. If an agent knows the class, it can predict the values of the other features. If it does not know the class, Bayes' rule can be used to predict the class given the feature values. In a Bayesian classifier, the learning agent builds a probabilistic model of the features and uses that model to predict the classification of a new example.

The simplest case is the naive Bayesian classifier, which makes the independence assumption that the input features are conditionally independent of each other given the classification. The independence of the naive Bayesian classifier is embodied in a particular belief network where the features are the nodes, the target variable (the classification) has no parents, and the classification is the only parent of each input feature. This belief network requires the probability distributions *P(Y)* for the target feature Y and $P(X_i \mid Y)$ for each input feature X_i. For each example, the prediction can be computed by conditioning on observed values for the input features and by querying the classification[16].

Given an example with inputs $X_1 = v_1$, ..., $X_k = v_k$, Bayes' rule is used to compute the posterior probability distribution of the example's classification, Y:

$$P(Y \mid X_1 = v_1, ..., X_k = v_k)$$
$$= \frac{P(X_1 = v_1, ..., X_k = v_k \mid Y) \times P(Y)}{P(X_1 = v_1, ..., X_k = v_k)}$$

$$= \frac{P(X_1 = v_1 \mid Y) \times ... \times P(X_k = v_k \mid Y) \times P(Y)}{\sum_Y P(X_1 = v_1 \mid Y) \times ... \times P(X_k = v_k \mid Y) \times P(Y)}$$

where the denominator is a normalizing constant to ensure the probabilities sum to 1. The denominator does not depend on the class and, therefore, it is not needed to determine the most likely class.

To learn a classifier, the distributions of $P(Y)$ and $P(X_i \mid Y)$ for each input feature can be learned from the data. The simplest case is to use the empirical frequency in the training data as the probability (i.e., use the proportion in the training data as the probability). However, as shown below, this approach is often not a good idea when this results in zero probabilities.

Although there are some cases where the naive Bayesian classifier does not produce good results, it is extremely simple, it is easy to implement, and often it works very well. It is a good method to try for a new problem.

In general, the naive Bayesian classifier works well when the independence assumption is appropriate, that is, when the class is a good predictor of the other features and the other features are independent given the class. This may be appropriate for natural kinds, where the classes have evolved because they are useful in distinguishing the objects that humans want to distinguish. Natural kinds are often associated with nouns, such as the class of dogs or the class of chairs.

A class' prior may be calculated by assuming probable classes (i.e., priors = 1 / (number of classes)), or by calculating an estimate for the class probability from the training set (i.e., (prior for a given class) = (number of samples in the class) / (total number of samples)). To estimate the parameters for a feature's distribution, one must assume a distribution or generate non-parametric models for the features from the training set.

The assumptions on distributions of features are called the event model of the Naive Bayes classifier. For discrete features like the ones encountered in document classification (include spam filtering), multinomial and Bernoulli distributions are popular. These assumptions lead to two distinct models, which are often confused[31].

1. Gaussian naive Bayes

   When dealing with continuous data, a typical assumption is that the continuous values associated with each class are distributed according to a Gaussian distribution. For example, suppose the training data contain a continuous attribute x. We first segment the data by the class, and then compute the mean and variance of x in each class. Let μ_c be the mean of the values in x associated with class c, and let σ_c^2 be the variance of the values in associated with class c. Then, the probability distribution of some value given a class, $p(x = v|c)$, can be computed by plugging into the equation for a Normal distribution parameterized by μ_c and σ_c^2. That is,

   $$p(x = v|c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(v-\mu_c)^2}{2\sigma_c^2}}$$

   Another common technique for handling continuous values is to use binning to discretize the feature values, to obtain a new set of Bernoulli-distributed features; some literature in fact suggests that this is necessary to apply naive Bayes, but it is not, and the discretization may throw away discriminative information.[32]

2. Multinomial naive Bayes

With a multinomial event model, samples (feature vectors) represent the frequencies with which certain events have been generated by a multinomial $(p_1, ..., p_n)$ where p_i is the probability that event i occurs (or k such multinomial in the multiclass case). A feature vector $x = (x_1, ..., x_n)$ is then a histogram, with x_i counting the number of times event i was observed in a particular instance. This is the event model typically used for document classification, with events representing the occurrence of a word in a single document. The likelihood of observing a histogram x is given by

$$p(x|C_k) = \frac{(\Sigma_i x_i)!}{\prod_i x_i!} \prod_i p_{k_i}^{x_i}$$

The multinomial naive Bayes classifier becomes a linear classifier when expressed in log-space:[33]

$$logp(C_k|x) \alpha log(p(C_k) \prod_n^{i=1} p_{k_i}^{x_i}$$

$$= logp(C_k) + \sum_{i=1}^{n} x_i \cdot logp_{k_i}$$

$$= b + W_k^T X$$

where $b = logp(C_k)$ and $w_{k_i} = logp_{k_i}$.

If a given class and feature value never occur together in the training data, then the frequency-based probability estimate will be zero. This is problematic because it will wipe out all information in the other probabilities when they are multiplied. Therefore, it is often desirable to incorporate a small-sample correction, called pseudo-count, in all probability estimates such that no probability is ever set to be exactly zero. This way of regularizing naive Bayes is called Laplace smoothing when the pseudo-count is one, and Lidstone smoothing in the general case.

The advantages and disadvantages of Bayes classifier as follows:

- Fast to train (single scan)
- fast to classify
- Not sensitive to irrelevant features
- Handles real and discrete data

- Handles streaming data well
- Assumes independence of features

2.3.3 Classification based on association rule

Association rule mining is an important task for discovering interesting relations between variables in large databases. It is a strong tool to discover the rules in data mining[34]. Association rule mining is presented by Agrawal, Imielinski and Swami in their paper in 1993 [35]. It aims to investigate the shopping habits of customers to find regularities.

The prototypical application is market basket analysis, that is, to mine the sets of items that are frequently bought together at a supermarket by analyzing the customer shopping carts(the so-called market baskets). Once we mine the frequent sets, they allow us to extract association rules among the item sets, where we make some statement about how likely are two sets of items to co-occur or to conditionally occur. In addition to the above market basket analysis, association rules are employed today in many application areas including Web usage mining, intrusion detection, continuous production, and bioinformatics. For example, in the web log scenario frequent sets allow us to extract rules like, "users who visit the sets of pages main, laptops and rebates also visit the pages shopping-cart and checkout", indicating, perhaps, that the special rebate offer is resulting in more laptop sales. In the case of market baskets, we can find rules such as "Customers who buy milk and cereal also tend to buy bananas", which may prompt a grocery store to co-locate bananas in the cereal aisle. In contrast with sequence mining, association rule learning typically does not consider the order of items either within a transaction or across transactions.

Definition Let $I = \{i_1, i_2, ..., i_n\}$ be a set of binary attributes called items. Let $D = \{t_1, t_2, ..., t_m\}$ be a set of transactions called the database. Each transaction in D has a unique transaction ID and contains a subset of the items in I. A *rule* is defined as an implication of the form $X \Rightarrow Y$, where $X, Y \subseteq I$ and $X \cap Y = \emptyset$. The sets of items (for short item sets) X and Y are called antecedent (left-hand-side or LHS) and consequent (right-hand-side or RHS) of the rule respectively. [35]

To illustrate the concepts, we use a small example from the supermarket domain. The set of items is $I = \{milk, bread, butter, beer, diapers\}$ and in the table to the right is shown a small database containing the items (1 codes presence and 0 codes absence of an item in a transaction) which is called binary dataset[35]. An example rule for the supermarket could be $\{butter, bread\} \Rightarrow \{milk\}$ meaning that if butter and bread are

Table 2.1: Binary database

Example database with 5 items					
Transaction ID	Milk	Bread	Butter	Beer	Diapers
1	1	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	1
4	1	1	1	0	0
5	0	1	0	0	0

Table 2.2: Transaction database

Example database with 5 items	
Transaction ID	Items
1	Milk Bread
2	Butter
3	Beer Diapers
4	Milk Bread Butter
5	Bread

bought, customers also buy milk. [35]

To select interesting rules from the set of all possible rules, constraints on various measures of significance and interest can be used. The best-known constraints are minimum thresholds on support and confidence.

- The support *supp(X)* of an item set X is defined as the proportion of transactions in the database which contain the item set. In the example database, the item set {milk, bread, butter}has a support of 1/5=0.2 since it occurs in 20% of all transactions (1 out of 5 transactions). The argument of *supp()* is a set of preconditions, and thus becomes more restrictive as it grows (instead of more inclusive).

- The confidence of a rule is defined as $conf(X \Rightarrow Y) = supp(X \cup Y)/supp(X)$. For example, the rule $\{butter, bread\} \Rightarrow \{milk\}$ has a confidence of 0.2/0.2=1 in the database, which means that for 100% of the transactions containing butter and bread the rule is correct (100% of the times a customer buys butter and bread, milk is bought as well). Note that $supp(X \cup Y)$ means the support of the union of the items in X and Y. This is somewhat confusing since we normally think in terms of probabilities of events and not sets of items. We can rewrite $supp(X \cup Y)$ as the joint probability $P(E_X \cap E_Y)$, where E_X and E_Y are the events that a transaction

contains item set X or Y, respectively.[36] Thus confidence can be interpreted as an estimate of the conditional probability , the probability of finding the RHS of the rule in transactions under the condition that these transactions also contain the LHS.

- The lift of a rule is defined as $lift(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X) \times supp(Y)}$ or the ratio of the observed support to that expected if X and Y were independent. The rule $\{milk, bread\} \Rightarrow \{butter\}$ has a lift of $\frac{0.2}{0.4 \times 0.4} = 1.25$.

- The conviction of a rule is defined as $conv(X \Rightarrow Y) = \frac{1 - supp(Y)}{1 - conf(X \Rightarrow Y}$. The rule $\{milk, bread\} \Rightarrow \{butter\}$ has a conviction of $\frac{1-0.4}{1-0.5} = 1.2$, and can be interpreted as the ratio of the expected frequency that X occurs without Y (that is to say, the frequency that the rule makes an incorrect prediction) if X and Y were independent divided by the observed frequency of incorrect predictions. In this example, the conviction value of 1.2 shows that the rule $\{milk, bread\} \Rightarrow \{butter\}$ would be incorrect 20% more often (1.2 times as often) if the association between X and Y was purely random chance.

Other types of association mining

Multi-Relation Association Rules: Multi-Relation Association Rules (MRAR) is a new class of association rules which in contrast to primitive, simple and even multi-relational association rules (that are usually extracted from multi-relational databases), each rule item consists of one entity but several relations. These relations indicate indirect relationship between the entities. Consider the following MRAR where the first item consists of three relations live in, nearby and humid: Those who live in a place which is near by a city with humid climate type and also are younger than 20 -¿ their health condition is good. Such association rules are extractable from RDBMS data or semantic web data.[37]

Context Based Association Rules is a form of association rule. Context Based Association Rules claims more accuracy in association rule mining by considering a hidden variable named context variable which changes the final set of association rules depending upon the value of context variables. For example the baskets orientation in market basket analysis reflects an odd pattern in the early days of month.This might be because of abnormal context i.e. salary is drawn at the start of the month.

Contrast set learning is a form of associative learning. Contrast set learners use rules that differ meaningfully in their distribution across subsets.[26][27] Weighted class learning is another form of associative learning in which weight may be assigned to

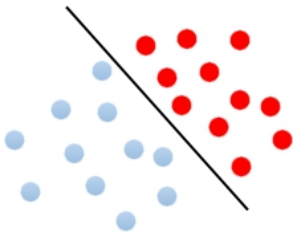


Figure 2.3: Linear classifier

classes to give focus to a particular issue of concern for the consumer of the data mining results.

High-order pattern discovery facilitate the capture of high-order (polythetic) patterns or event associations that are intrinsic to complex real-world data.

Sequential pattern mining discovers subsequences that are common to more than minsup sequences in a sequence database, where minsup is set by the user. A sequence is an ordered list of transactions.

2.3.4 Support vector machine

Support Vector Machines (SVMs) is a classification method based on maximum margin linear discriminants, that is, SVMs are based on the concept of decision planes[38]. The goal is to find the optimal hyperplane that maximizes the gap or margin between the classes. A decision plane is one that separates between a set of objects having different class memberships. A schematic example is shown in the illustration figure 2.3. In this example, the objects belong either to class BLUE or RED. The separating line defines a boundary on the right side of which all objects are BLUE and to the left of which all objects are RED. Any new object (white circle) falling to the right is labeled, i.e., classified, as BLUE (or classified as RED should it fall to the left of the separating line).

The figure 2.3 is a classic example of a linear classifier, i.e., a classifier that separates a set of objects into their respective groups (BLUE and RED in this case) with a line. Most classification tasks, however, are not that simple, and often more complex structures are needed in order to make an optimal separation, i.e., correctly classify new objects (test cases) on the basis of the examples that are available (train cases). This situation is depicted in the illustration figure 2.4. Compared to the previous schematic, it is clear that a full separation of the BLUE and RED objects would require a curve (which is more complex than a line). Classification tasks based on drawing separating lines to distinguish between objects of different class memberships are known as hyperplane classifiers. Support Vector Machines are particularly suited to handle

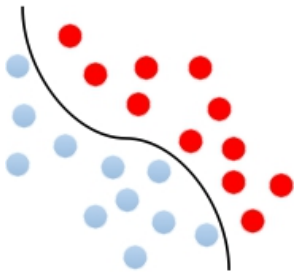


Figure 2.4: Hyperplane classifier

such tasks.

Support Vector Machine (SVM) is primarily a classier method that performs classification tasks by constructing hyperplanes in a multidimensional space that separates cases of different class labels. SVM supports both regression and classification tasks and can handle multiple continuous and categorical variables. For categorical variables a dummy variable is created with case values as either 0 or 1. Thus, a categorical dependent variable consisting of three levels, say (A, B, C), is represented by a set of three dummy variables:

A: {0 0 1}, B: {0 1 0}, C: {1 0 0}

To construct an optimal hyperplane, SVM employs an iterative training algorithm, which is used to minimize an error function. According to the form of the error function, SVM classification models can be classified into two distinct groups:

Classification SVM Type 1 (also known as C-SVM classification)

For this type of SVM, training involves the minimization of the error function:

$$\frac{1}{2}w^T w + C\sum_{i=1}^{N}\xi_i$$

subject to the constraints:

$$y_i(w^T\phi(x_i) + b) \geq 1 - \xi_i \quad and \quad \xi_i > 0, i = 1, ..., N$$

where C is the capacity constant, w is the vector of coefficients, b is a constant, and ξ_i represents parameters for handling nonseparable data (inputs). The index i labels the N training cases. Note that $y \in \pm 1$ represents the class labels and x_i represents the independent variables. The kernel ϕ is used to transform data from the input (independent) to the feature space. It should be noted that the larger the C, the more the error is penalized. Thus, C should be chosen with care to avoid over fitting.

Classification SVM Type 2 (also known as nu-SVM classification)

In contrast to Classification SVM Type 1, the Classification SVM Type 2 model minimizes the error function:

$$\frac{1}{2}w^T w - v\rho + \frac{1}{N}\sum_{i=1}^{N}\xi_i$$

subject to the constraints:

$$y_i(w^T\phi(x_i) + b) \geq \rho - \xi_i, \xi_i \geq 0, i = 1, ..., N \quad and \quad \rho > 0$$

2.3.5 Genetic algorithm

Genetic algorithms(GA) is adaptive heuristic search algorithm based on the evolutionary ideas of natural selection and genetics in the field of artificial intelligence. It is proposed by Holland in 1975[94]. The basic technique of the genetic algorithm is designed to simulate processes in natural systems necessary for evolution. This algorithm is usually used to generate useful solutions to optimization and search problems. It exploits historical information to direct the search into the region of better performance within the search space.

Genetic algorithms simulate the survival of the fittest among individuals over consecutive generation for solving a problem. Each generation consists of a population of character strings that are analogous to the chromosome. Each individual represents a point in a search space and a possible solution. The individuals in the population are then made to go through a process of evolution.

The basic operation process of genetic algorithm is as follows:

a) Initialization: Setting evolution generation counter t = 0, set the maximum evolution generation T, M individuals randomly generated as initial population P (0).

b) Individual evaluation: calculating the fitness of each individual in population P (t).

//A fitness score is assigned to each solution representing the abilities of an individual to 'compete'.

c) Selection operation: the purpose is choosing optimal individuals or new individuals produced by paring and crossing into the next generation. Selection operation is based on the assessment of the fitness of individuals in a population.

d) Crossover operation: crossover operator play important role in genetic algorithms.

e) Mutation operation: to change the genetic value of certain individual strings in the population. Population P (t) evolves into the next generation of population P (t + 1) through selection, crossover and mutation operation.

f) Termination condition: if t = T, output the optimal solution that the individual with a maximum fitness, terminate the calculation.

The flow chart of genetic algorithm is shown in Figure 2.5.

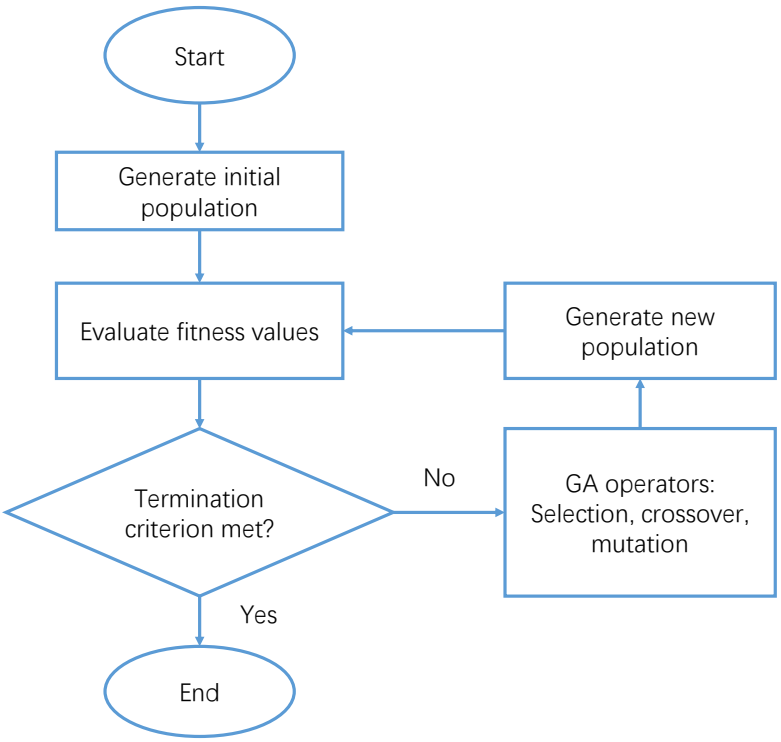


Figure 2.5: Genetic algorithm flow chart

The characteristics of genetic algorithm are below:

- Operate directly on the structure of the object, and the continuity of function derivative is defined does not exist.
- Global implicit inherent parallelism and better optimization capabilities.
- Probabilistic method of optimization that can automatically obtain and guide optimized search space adaptively adjust the search direction, the rule does not require determined.

There are limitations of the genetic algorithm:

- Repeated fitness function evaluation for complex problems is often the most prohibitive and limiting segment of artificial evolutionary algorithms. Finding the optimal solution to complex high-dimensional, multi-modal problems often requires very expensive fitness function evaluations.

- Genetic algorithms do not scale well with complexity. That is, where the number of elements which are exposed to mutation is large there is often an exponential increase in search space size. This makes it extremely difficult to use the technique on problems such as designing an engine, a house or plane. In order to make such problems tractable to evolutionary search, they must be broken down into the simplest representation possible.

- In many problems, genetic algorithm may have a tendency to converge towards local optima or even arbitrary points rather than the global optimum of the problem. This means that it does not "know how" to sacrifice short-term fitness to gain longer-term fitness.

- Operating on dynamic data sets is difficult, as genomes begin to converge early on towards solutions which may no longer be valid for later data.

- Genetic algorithm cannot effectively solve problems in which the only fitness measure is a single right/wrong measure (like decision problems), as there is no way to converge on the solution (no hill to climb).

- For specific optimization problems and problem instances, other optimization algorithms may be more efficient than genetic algorithms in terms of speed of convergence.

2.3.6 Analytic hierarchy process

Analytic Hierarchy Process(AHP) is a structured decision-making technique to decompose the decision-making related elements to goals, guidelines, programs and other levels in order to make qualitative and quantitative analysis. It was first proposed by Thomas Saaty [95] in the 1970s and then is used widely in many decision environments. Instead of providing a correct decision, the analytic hierarchy process try to find the best suitable decision that is consistent with the understanding of decision makers. To use the analytic hierarchy process, the decision makers need first decompose the decision problem into many independent sub-problems. In the decision making process, decision makers can take part in the process by making their own judgements. It means the subjective judgements of individuals can have a great influence on the decision making process.

The decision-making process for analytic hierarchy process is as follows:

1. Model the decision problem as a hierarchy. Specify the decision goal, the alternatives, and the criteria.

2. Establish priorities among the elements of the hierarchy by making a series of judgements based on pairwise comparisons of the elements.

3. Synthesize these judgements to yield a set of overall priorities for the hierarchy.

4. Check the consistency of the judgements.

5. Come to a final decision based on the results of this process.

The advantages of analytic hierarchy process are listed as follows.

1. First, it is a systematic analysis method. The analytic hierarchy process takes the decision problems as a system. The final result is affected by all the factors in the system. The weights in each layer of the system will directly or indirectly affect the final result. This method is suitable for evaluation of multi-objective, multi-criteria and multi-period system.

2. Second, it is quite simple and easy to use. It transforms the multi-goals problems into multi-hierarchy with single goal problems, which can greatly simplify the computation. It is easy for decision makers understand.

3. Third, it needs less quantitative information. It simulates the way of how people make decisions by leaving important information for brains. This can simplify the calculate overhead and solve many practical problems that cannot be solved by conventional optimizing problems.

The disadvantages of analytic hierarchy process include:

1. First, it cannot provide new decision-making policy. The analytic hierarchy process is used to select the best policy form the candidates. All the policies are known before. The analytic hierarchy process is not able to propose new policy different form the candidates.

2. Second, many qualitative factors make it hard to believe. It introduces many qualitative factors by simulating the decision-making process of human brains.

3. Third, the statistics grows with the criterion.

The analytic hierarchy process is quite useful for groups encountering the complex problems. It can tackle the decision problem well even if the important elements of the decision are missed. The analytic hierarchy process has been widely used in complex decision situations. It can be applied in the following situations. First one is choice, the analytic hierarchy process is used to select the best policy from a set of candidates. Second one is similar to choice, called ranking. It sorts all the candidates according to some criterion. Third is quality management. The analytic hierarchy process measures the different aspects of quality.

2.4 The challenges of cloud service selection

Cloud service selection is the one includes very wide-ranging topic for discussion. In distributed and constantly changing cloud computing environments there are many challenges, such as (i) automated recommended system of service selection constantly matching the appropriate service according to user requirements, (ii) to promptly satisfy incoming cloud user requirements in cloud service composition, collaboration between brokers and service providers is necessary, (iii) ranking multiple services or optimizing services composition are also key issues, (iv) determining the importance of parameters of cloud services and selecting cloud service providers. Figure 2.6 is the process of cloud service requesting, binding, delivery. Available single cloud service or cloud service composition on the worldwide service pool published by cloud service provider are introduced to the broker, who according to the users' requirements or intention to select the best service or set of services to users.

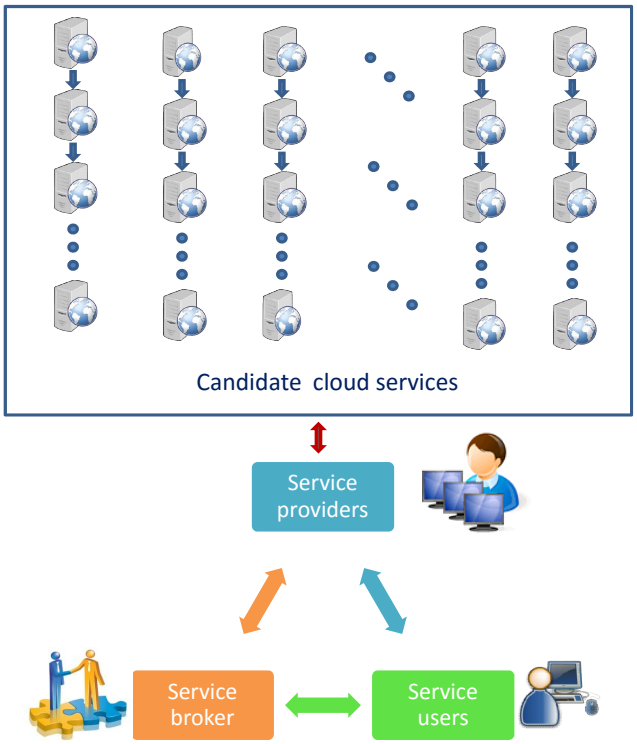


Figure 2.6: Cloud service selection process of requesting, binding, delivery

2.4.1 Cloud service composition

With the development in the utilization of cloud computing, more and more similar-function services increase for different servers. These similar services with distinct values in terms of the Qos(quality of service) parameters are distributed in different locations. Service composition techniques aim to select multiple atomic services with different function among the similar services that are located on different servers to composite the set of cloud services to allow the highest Qos to be achieved according to the users' demands and priorities. The available services and demands of the user are constantly changing in cloud environments, service composition technique should have automated function capabilities to accommodate it. Therefore, selecting appropriate and optimal simple services to composite together to provide set of services, namely service composition, is one of the most important problems in cloud service selection. The researchers have done a lot of studies, with cloud computing technique development, it always brings many new challenges about selection and composition of services in dynamic cloud environment, this needs more researchers to provide the corresponding solutions.

2.4.2 Cloud service composition problem challenges

More and more cloud service users using cloud computing encourages cloud service providers to supply services with different functional and nonfunctional features in a service pool. Cloud service requirements can be mapped to cloud resources in an automated manner. The cloud service composition pattern need consciously changes in a period of time due to the dynamic characteristics of cloud environments. This causes a series of challenges to the service composition. The major challenges are the following:

Cloud providers supplying elastic service. Most service providers making pricing policy of the cloud service charge is based on supply and demand. Thus, there should have an available mechanisms to predict and manage the renewable resources[49].

Dealing with incomplete cloud resources. The information integrity of services is the guarantee of optimal service selection by a broker[49][50].

Designing multi-cloud application in cloud platform. Various platforms offer facilities for single cloud application design, deployment and provisioning, there also should platforms to design and deploy multiple clouds application for selecting the best possible cloud service composition based on user requirement[55].

Inter-service composition restriction. Dependency or conflicts between two or more

services results in a complicated service composition problem. In selecting service composition, dependency and conflict among services is quite common and can not be ignored [51].

2.4.3 Existing cloud service composition works

In recent years, cloud computing technology grows quickly, which is evolving as a widely used computing platform where many different web services are published and available in cloud computing centers. Single service could not completely fulfill the user requirements, it is necessary to compose the functionalities of multiple web services, the process of compose the services is called "*service composition*"[57]. The process of service composition should consider a set of end-to-end Qos constraints(local and global) raised by users and find an optimal composite solution to satisfy the users' requirements. Service composition algorithms try to find a global optimal composite solution, the process of service composition is to be considered an NP-hard problem due to huge search space. Zeng et [59] present a middle-ware platform which handles the issue of selecting the set of web service composition in a way that satisfying the constraints set by the suer and by the structure of the composite service. [58] Alrifai et Risse combine global optimization with local selection techniques to support rapid and dynamic service compositions.

ABC(artificial bee colony) are widely adopted to find an approximately optimal solution in the restricted condition. In literature [53], the work focuses on improvement of traditional ABC neighborhood strategy for local search, with the objective of better optimality and faster convergence rate. The authors proposed approximate-Mapping Von Neumann algorithm(AMV). Firstly, the discrete spaces of service composition problem are approximately transformed to a continuous space in which a locally optimal neighboring solution is precisely found due to traditional ABC is good at dealing with service composition problem in a continuous space. Secondly, they adopt the Von Neumann neighborhood topology to further improve the quality of local search.

In literature[20], the researchers added time attenuation function into the service composition model, thus service composition is transformed into a nonlinear integer programming problem. The Discrete Gbest-guided Artificial Bee Colony algorithm proposed simulates the search for optimal service composition solution through the exploration of bees for food. For the large-scale data, it can obtain a near-optimal solution with less time.

Cloud manufacturing takes advantage of cloud computing technique, information

technology and advanced management technologies et to build the collaboration among different organizations to make full of various manufacturing resources. Optimizing the optimal resources allocation is critical in manufacturing cloud service composition. The paper[60] presented a correlation-aware manufacturing cloud service description model to characterize the Qos dependence between cloud services. Based on it, the authors proposed a service correlation mapping model for getting correlation Qos values among cloud services automatically. Furthermore, an effective service selection approach is proposed based on a genetic algorithm.

In most of researches, service composition methods take a hypothesis that all selected cloud services found in the composition sequence storing in one service repository, rather than those cloud services distributed in different locations. It is a challenge to efficiently find a composite solution in a multiple cloud base due to the distributed and diversification features. For this problem, [56] Zou et first propose a framework of service composition in multi-cloud base environment. Next, the authors proposed a cloud combination method based on artificial intelligence planning which not only finding feasible composition sequence, but also containing minimum clouds, it is effective to find sub-optimal cloud combinations.

An increasing interest in web service composition shift from a single cloud to multi cloud because of its importance in practical applications. The available approaches generating composite service in a single cloud, which limits the benefits that are derived from other clouds. Literature [61] proposes a novel COMbinatorial optimization algorithm for cloud service COMposition(COM2)that can effective use of multiple clouds, and which ensures that the cloud with the maximum number of services will always be selected before other clouds and increases the possibility of fulfilling service requests with minimal overhead.

Cloud computing and big data have attracted much attention from both academic and industry communities. Cloud computing promises a scalable infrastructure and software platform for processing big data applications. In practice, certain big data centers cannot be transplanted into a public cloud due to some security and privacy. Specially, some privacy clouds refuse to disclose their service transaction records because of business privacy in cross-cloud scenarios. To overcome this challenge, [62] Dou et propose a privacy-aware cross-cloud service composition approach, named History record-based Service optimization method(HireSome-II) which aims to enhance the credibility of a composition framework and evaluate the services by its Qos history records. In this approach, the authors introduced the k-means algorithm as a data filtering tool to select representative history records.

Cloud composition optimal-selection(SCOS) is a typical NP-hard problem because of the characteristics of dynamic and uncertainty. The traditional methods for solving large scale SCOS problem with numerous constraints is not inefficient in cloud manufacturing system. To overcome this shortcoming, Huang et [63] propose a novel parallel intelligent algorithm, named full connection based parallel adaptive chaos optimization with reflex migration. The algorithm combining the virtues of the adaptation of chaotic sequences and roulette wheel selection is designed for high quality decision in series. To improve the searching efficiency further, the algorithm adopts full connection topology based on coarse-grained parallelization and MPI (Mean Point of Impact) collective communication.

In cloud environment, for a given service request, there could be a large number of software service meeting the functional requirements. A software service might need collaboration from other types of cloud service to provide a solution to a cloud user, there should have a way to measure the whole solution. In literature 64], researchers proposed a model for predicting end-to-end QoS values of cloud service compositions in a cloud-based service selection system, which relies on the internal features of services and cloud users such as locations, functionality and preference requirements to compute service matching value.

Service composition enables us to reuse existing services with less cost and time consumption. One of the problems for service composition is to maximize the overall Qos of the composite service. Some researchers have done a lot of works in a little different emphasis, but their aims are the same that selecting optimal set of service composition to satisfy the uses' requirements. CANFORA et al [32] and Li et al [36] respectively proposed an approach for Qos-aware service composition based on genetic algorithm, the difference of approaches is that the later applies in multi-networks. To improve the efficiency of the service composition, Liu et al[34] proposed an improved genetic algorithm to solve Qos-aware service composition problem, which combines Ant Colony Optimization and Genetic Algorithm. Yilmaz et al[33] proposed an approach based on improved genetic algorithm to optimize the overall Qos of service composition.

In a service composition, optimizing some Qos attributes under given Qos constraints has been shown to be NP-hard. Heuristic algorithm is widely used to find acceptable solutions in polynomial time. However, heuristic algorithm usually has a high time complexity for real-time use until it finds near-optimal solutions. At this point, KLEIN et al[39] proposed an efficient heuristic approach with improved time complexity for Qos-aware service composition, which is based on Hill-Climbing with a greatly reduced search space that makes effective use of an initial bias computed

with linear programming to have a much lower complexity. Furthermore, the approach obtains near-optimal solutions in just a fraction of the time required for the standard Hill-Climbing algorithm.

Service-oriented architecture realize the composition of loosely coupled services provided with varying Qos levels. However, in a business environment, there are additional requirements for service compositions such as a high reliability. Thus, in contrast to traditional service composition, literature[37] proposes a holistic probabilistic approach that is tailored to long-term service composition problem in business-to-business(B2B) environment. The approach using Qos pattern, usage pattern and time-dependent invocation policies can select the most appropriate services and backup services for some specific users. For this purpose, authors introduce an adaptive heuristic algorithm based on genetic algorithm, which adjusts the number of backup services to the reliability constraint of the user.

Bao et al[72] proposed a method to model web services by using Finite State Machine(FSM), to address the problem that the service constraints in the cloud environment. The scheme of this method consists of two steps. Firstly, the researchers introduced an improved Tree-pruning-based algorithm to build the composition tree, at the same time, marking each path to avoid traversing the tree again, which greatly reduce the execution time of the algorithm. Then adopting a simple additive weighting technique to select an optimal service. WU et al[71] proposed a service composition topology reconfiguration model for multi-site service composition application in mobile cloud computing environment. In this model, multiple surrogates, such as cloud computing nodes, mobile devices and their services can be composed to fulfill tasks required by mobile users.

In cloud service composition, collaboration between brokers and service providers is very important to promptly meet incoming cloud users' requirements. User requirements should be satisfy via web services via web services in an automate manner. However, cloud computing environments are distributed and constantly changing, this needs contracting dynamically between service user and service provider. To solve this issues, in literature[40], Gutierrez-garcia et al proposed an agent-based cloud service composition approach. The main idea is that, firstly, the self-organizing agents make use of acquaintance networks to cope with partial information of cloud computing environments and contract net protocol to evolve and adapt cloud service composition.

2.4.4 Existing other cloud service selection works

Cloud computing technique offers great opportunities for companies or institutions to share the IT resources with the best service and pricing, there are some challenges on how to select the best service or service provider in the huge resource pool. It is a very time-consuming for users to collect the necessary information and analyze all service providers to make decisions. For this problem, Sundareswaran et al.[20] proposed a novel brokerage-based framework in the cloud, where cloud brokers help cloud users select and rank the cloud service providers based on the users' requirements. For the service selection approach, the authors design a unique indexing technique for managing the information of a large number of cloud service providers.

In literature [76], Badidi Elarbi proposed a framework for SaaS (Software-as-a-Service) provisioning, the cooperation between cloud user and cloud service provider based on SLAs(Service Level Agreements). A cloud service broker helps cloud users select the appropriate SaaS provider that can fulfill users' functional and Qos requirements. In additional, the cloud service broker is in charge of the negotiating the SLAs with the provider selected on behalf of the cloud users, and monitoring the compliance to the SLAs during its implementation.

With the technological advancements, an industrial economy transforms into an information economy gradually. Most enterprises together via advanced information network technique to share resources in order to fulfill a specific business task. For service oriented enterprises, they encapsulate the computing resources as service and published online. As there are more and more available services providing similar functionalities but different potential business correlations between them, it brings some challenges for selecting the services. For this problem, Wu et al. [77] proposed a business correlation model of service selection correlations. Then they give an efficient approach for correlation-driven QoS-aware optimal service selection based on a genetic algorithm.

Cloud service providers such as IBM, Microsoft, Google, and Amazon offer different cloud services to their users. It has become difficult for users to decide whose services are appropriate and what is the standard for their selection. For this issue, Garg et al. [78] propose a framework and a mechanism to measure the quality and rank the cloud services. This framework is good to both cloud users and providers, because it can help cloud users make a decision and cloud provider improve their service quality.

Literature [79] presented a general optimization framework to solve the data-center selection problem for cloud services. the authors proposed a distributed algorithm based on the sub-gradient through a dual decomposition approach. Literature [80] describes

a mechanism in which context is gathered relation to service providers. It can be used for Service-oriented Architectures to select appropriate service providers.

With the cloud service development, more and more enterprise published their computing resources encapsulated as services online. Reputation mechanism is necessary to establish trust on prior unknown services. In literature[81], the researchers proposed a improved reputation bootstrapping approach, the advantage of this approach is that it can give default reputation value for newcomers. The main idea of the approach is that it can establish a tentative reputation for new or unknown services according to correlation generalised between features and performance of existing services are learned through an artificial neural network.

Mobile cloud computing can deal with issues by executing mobile applications on resource providers external to the mobile device. However, selecting the appropriate server no service delay for mobile device is difficult. For this problem, Liu et al. [82] proposed a mobility-aware framework for mobile cloud streaming services, which provides dynamic and optimized service selection functions to support user mobility comprehensively with less service delay and high service quality, it makes service selection scheme suitable for mobile environment.

For selecting and composing web services problems, literature [83] introduces firstly transactional properties of a single web service and the transactional rules used to compose the services, then it proposes a genetic algorithm which takes into consideration the execution time, price, transactional property, stability, and penalty-factor to achieve globally optimal service selection. Finally, this paper gives the result of experiments that compare the proposed approaches such as transactional Qos driven selection algorithm and exhaustive search algorithm.

RUIZ-ALVAREZ et al[84] propose an automated approach to select the cloud storage service which depends on a machine readable description of the capabilities of each storage system that can meet the user's specific requirements. First, the authors present an XML schema based on the documentation of different storage services to provide descriptions for the cloud storage service system such as Amazon Azure and local clouds. Then, they develop an application that processes XML descriptions to match common data requirements from users. The main achievement of this paper is able to recommend storage services for a cloud application, estimate storage costs and performance under different growth scenarios and provide information to assist in migrating the cloud application to private cloud deployments.

OLIVEIRA et al [85] present a research model based on the innovation characteristics from the diffusion of innovation theory and the technology-organization-environment

framework to assess the determinants that influence the adoption of cloud computing. The model was empirically evaluated based on a sample of 369 firms in Portugal. This study shows that in evaluating the adoption of cloud computing that takes into consideration the technology, organization, and environment contexts of the organization along with the innovation characteristics is more holistic and meaningful in providing valuable insights to practitioners and researchers.

Cloud service selection in a multi-cloud environment increasingly attracts the attention of researchers. It's hard for users to select an appropriate service for their application in a dynamic multi-cloud environment, especially for online real-time applications. To help users to efficiently select cloud service, the paper[86] develops a cloud service selection model adopting the cloud service brokers, based on it, a dynamic cloud service selection strategy is proposed. The cloud service selection strategy uses an adaptive learning mechanism that comprises the incentive, forgetting and degenerate function to dynamically optimize the cloud service selection process and the best service feedback to users.

The paper[87] proposes service selection optimization framework for balancing cost and benefits of private and public cloud in hybrid cloud computing. Hybrid cloud service selection optimization is distributed to and performed at hybrid cloud service and resource layers, it maximizes the interests of hybrid cloud user agents, hybrid cloud service agent, public cloud agents and private cloud agents. Hybrid cloud service selection process consists of two parts: hybrid cloud service provisioning and cloud resource allocation. The author presents a two-level hybrid cloud service selection algorithm used to perform service provisioning and resource allocation.

Zhang, Miranda, et al[88] present a PhD thesis proposal on investigating an intelligent decision support system for selecting Cloud-based infrastructure services. They identify the following hard research issues in the domain of cloud service selection and comparison. Question 1, Automatic service identification and representation. Question 2, Optimized Cloud Service Selection and Comparison. Question 3, Simplified interfaces for Cloud Service Selection. For question 1, authors presented a declarative approach to Cloud service selection, comparison and its implementation as Cloud Recommender system. To solve Q2, authors propose and develop a novel and flexible decision-making framework that builds upon two distinct techniques: i) evolutionary optimization techniques, the process of simultaneously optimizing two or more conflicting objectives expressed in the form of linear or nonlinear functions of criteria; ii) a decision making method, attempting to identify and select alternatives based on the value and the goals of decision makers. To solve Q3, authors investigate a widget-based

Table 2.3: Summary of approaches and characteristics considered by service selection

Reference	Approach	Qos-aware	Cloud environment	Framework
Kurdi et al.(2015)	GA	-	multi-cloud	yes
Jin et al.(2015)	GA	yes	single-cloud	yes
Dou et al.(2015)	Hiresome	yes	multi-cloud	-
Huang et al.(2014)	CCOA	yes	-	-
Karim et al.(2015)	-	yes	single-cloud	yes
Canfora et al.(2005)	GA	yes	-	no
Ylmaz et al.(2014)	IGA	yes	-	no
Liu et al.(2010)	IGA	yes	-	no
Klein et al.(2011)	HA	yes	-	no
Li et al.(2011)	GA	yes	multi-cloud	-
Klein et al.(2012)	PA	yes	-	no
Wu et al.(2015)	-	no	mobile-cloud	yes
Bao et al.(2012)	SAW	yes	single-cloud	yes
Gutierrez et al.(2010)	Agent-based	no	single-cloud	yes
Kritikos et al.(2015)	-	no	multi-cloud	yes
Huo et al.(2015)	ABC	yes	single-cloud	no
Min et al.(2014)	ABC	yes	-	no
Qi et al.(2013)	skyline	yes	-	yes
Jula et al.(2013)	ICA	yes	-	no
Karim et al.(2013)	AHP	yes	single-cloud	yes
Saripalli et al.(2011)	SAW/MADM	no	-	-
Wang et al.(2011)	CM	yes	single-cloud	-
Wu Quan et al.(2015)	Neural Network	-	-	no
Wang Xiao et al.(2015)	ALM	-	multi-cloud	-
Skoutas et al.(2010)	MCDR	no	-	yes
Liu Ran et al.(2015)	CQS3	-	mobile-cloud	yes
C.Y.Mao(2010)	RS	-	single-cloud	-
Liu et al.(2014)	RS	no	single-cloud	yes
W.WU(2010)	RS	no	-	-
Ghezzi et al.(2015)	PD	-	multi-cloud	-
Sun et al.(2013)	AHP	-	single-cloud	no

Approach: GA-genetic algorithm; Hiresome-history record-based service optimization method; CCOA-chaos control optimal algorithm; IGA-improved genetic algorithm; HA-heuristic approach; PA-probabilistic approach; SAW-simple additive weighting; ABC-artificial bee colony algorithm; ICA-imperialist competitive algorithm; AHP-analytic hierarchy process; ALM-adaptive learning mechanism; MADM-multiple attribute decision methodology; MCDR-multicriteria dominance relationship; CQS3-client driven QoS oriented server selection scheme; RS-rough set; PD-performance driven.

visual programming language to simplify the interaction with Cloud Services.

With the increasing of service providers, multiple providers may compete with each other by publishing services that provide the same functionality, but QoS and performance they offered are different. Users may dynamically select the most ef?cient services that satisfy their requirements among the competing alternatives. The paper [89] focuses on how to support users in performing dynamic binding to services. Firstly, authors formalize the service selection problem using a stochastic framework define a performance model of the users' experience. Then, they propose analyzing and comparing different service selection strategies.

The paper [90] propose a mono objective service selection approach based on harmony search algorithm. It can handle efficiently a large space of solutions, in order to find a near optimal composition that satisfies the QOS requirements and the end to end users constraints. [91] propose three ranking and clustering service algorithms based on the notion of dominance. [92] propose a novel approach based on iterative multi-attribute combinatorial auction that supports effective and efficient service selection.

Table 2.3 summarizes the main approaches, techniques and characteristics considered by cloud service selection. The researchers start the study with different emphasis, they provide the solutions for some specific problems. Thus, we summarize the four items: approach, Qos-aware, cloud environment and framework. To keep the name of study approach short, we use common abbreviation.

2.5 Conclusion

In this chapter, we stated the basic concepts and characters about cloud services. We present the related techniques and works of cloud service selection. In the next chapter, we will introduce the rough set theory as preliminary Knowledge.

Chapter 3

Related knowledge of rough set theory

3.1 Introduction

With the development of computer science and networks information technologies, data and information in various fields increase rapidly. As the involvement of the human, the uncertainty between data and information is more significant, the relations between them become complex. For abundant useful and available data and information resources, we are short of obtaining knowledge because we lack the effective mining methods to help us extract the useful information in big data. We should take full advantage of the data and information in database of small or large enterprises or institutions. Therefore, how to process the fuzzy, imprecise and incomplete big data to obtain potential, innovative and useful knowledge, it is a challenge.

Rough set theory and its method can effectively process data and information in complex system. It has become a new mathematical tool to process the fuzzy and imprecise problems. The obvious advantage of rough set theory compared with fuzzy set, evidence theory and probability theory methods for processing the uncertainty problems is that it needs not the priori information just data itself. In 1982, Z.Pawlak proposed the data analysis and reasoning theory - rough set. Initially the study of rough set theory was concentrated in eastern Europe, at that time, it didn't bring to attention. Until the early 1990s, rough set theory attracts wide concern from researchers in artificial intelligence and pattern recognition fields because it has been applied in data mining, decision analysis, machine learning and intelligent control successfully.

The main contexts of rough set theory are approximate classification, knowledge reduction(attributes or attributes values reduction), attributes dependency analysis,

getting an optimal or suboptimal decision control algorithm and so on. The study of rough set theory focuses on two aspects: one is the theory research, there are a series of literatures about rough set algebra, rough set topology and its properties, rough set logic, approximate reasoning and so on, which have formed system to process incomplete, imprecise, and uncertain problems; the other is application research, to study the rough set theory applied in many areas such as medical, management, image process, decision analysis and so on.

3.2 Rough set theory

rough set theory [116][118], introduced by Pawlak in the early 1980s, has become an important tool of soft computing. Rough sets has a strong qualitative analysis capability to express effectively uncertain or imprecise knowledge. It has been widely used in machine learning, rule generation, decision analysis, intelligent control, and other fields. Especially, it has a great success in the data mining domain. The main features of rough sets are strict mathematical definitions and robustness. Processing information with rough set theory on the basis of data does not require any additional prerequisites.

3.2.1 Information system

Definition 1 [116][118] Let $T = (U, A, V, f)$ be an information system, where $U = \{X_1, X_2, \ldots, X_n\}$ is the finite set of objects; $A = C \cup D$ is the set of attributes, C is a conditional attributes set, D is the decision attribute set; $V = \cup V_\alpha$, where V_α is the set of values of attributes $\alpha \in A$. f is an information function and denotes the map of $U \times A \longrightarrow V$, which assigns a value to each attribute of each object.

3.2.2 Knowledge and Knowledge space

Knowledge can be the summary for information processing, interpretation, selection and transformation. It can be also regarded as the set of proposition and regulation. On general, it is divided into illustrative, procedural and controlled knowledge. Illustrative knowledge provides the concepts and facts, for example, in an intelligent retrieval system, it illustrates the database for real facts; using rules to represent the problems is called procedural knowledge, usually, it is used to solve the illustrative knowledge in an intelligent retrieval system; controlled knowledge including all kinds of processing, strategies and structures to coordinate the solution for the whole problem. Here, we

primarily describe the knowledge pattern abstracted away from database with right, novel and potential application value to understand for people.

In rough set theory, knowledge is related with different classification pattern to real or abstract world. Any object can be described by knowledge. One can classify the objects according to the knowledge (various attributes or characteristics of objects). Knowledge is regarded as the classification ability for objects or knowledge itself, which can be represented by the set in knowledge system.

Definition 2 (Knowledge and concept)

Suppose U is the non-empty finite set of objects we are interested in, called an universe. Any subset $X \subseteq U$, called abstract knowledge respected to U.

The concept of an approximation space is used to describe certain analogies between spaces of sequences, functions and operations. The rough set theory is based on the concept of approximation space. Approximation spaces of an information system are defined by partition or coverings defined by attributes of a pattern space.

Definition 3 (Knowledge space)

Given an universe U and a cluster of equivalence relation S(it represents partition) in U, two-tuple $K = (U, S)$ is called as a knowledge base or approximation space.

3.2.3 In-discernibility relation

Definition 4 (In-discernibility relation)

Given an universe U and a cluster of equivalence relation S(it represents partition) in U, if $P \subseteq S$ and $P \neq \emptyset$, then $\cap P$ is also an equivalence relation in U, it is called the in-discernibility relation in P, denoted by $IND(P)$ or P. And that

$$\forall x \in U, [x]_{IND(p)} = [x]_P = \bigcap_{\forall R \in P} [x]_R$$

$U/IND(P) = \{[x]IND_{(P)} | \forall x \in U\}$ represents the knowledge related to the equivalence relation $IND(P)$, called P-basic set related to universe U in knowledge space $K = (U, S)$. Without confusion, P, U and K are clear, we can replace P with $IND(P)$ and $U/IND(P)$ with U/P. Equivalence classes of $IND(P)$ are called elementary categories of knowledge P.

3.2.4 Approximation space

Lower approximation sets and upper approximation set are used for the basic concepts of rough set theory. rough set theory analysis is based on two approximations. Lower and upper approximations are defined as following:

The lower approximation (3.1) and upper approximation (3.2) of the subset X about knowledge R are respectively defined by [116][118] as following,

$$\begin{aligned} \underline{R}(X) &= \{x|(\forall x \in U) \wedge ([x]_R \subseteq X)\} \\ &= \cup\{Y|Y \in U/R) \wedge (Y \subseteq X)\} \end{aligned} \tag{3.1}$$

$$\begin{aligned} \overline{R}(X) &= \{x|(\forall x \in U) \wedge ([x]_R \cap X \neq \emptyset)\} \\ &= \cup\{Y|Y \in U/R) \wedge (Y \cap X \neq \emptyset)\} \end{aligned} \tag{3.2}$$

Where, $[x]_R$ indicates an equivalence class of object x about knowledge R. U/R indicates elementary concepts of knowledge base K.

Set $Pos_R(X) = \underline{R}(X)$ is called positive region;

$Bn_R(X) = \overline{R}(X) - \underline{R}(X)$ is called boundary region;

$Neg_R(X) = U - \overline{R}(X)$ is called negative region.

Obviously, $\overline{R}(X) = Pos_R(x) \cup Bn_R(X)$.

The lower approximation set is the set of all objects of universe u certainly belonged to the set X on the universe U according to knowledge R; the upper approximation set consists of the lower approximation set and the objects of universe U cannot be ensured in the set X according to knowledge R. The boundary region $Bn_R(X)$ is consisted of the elements of universe U cannot be ensured in the set X according to knowledge R; The negative region $Ng_R(x)$ is consisted of the elements of universe U not in the set X according to knowledge R.

The lower and upper approximations of set X and boundary region shown in figure 3.1.

Example 3.1: In table 3.1 (a decision table), given a subset $X = \{e_2, e_3, e_5\}$ in universe U , for an attribute subset(equivalence relation) $P = \{headaches, muscular\ pains\}$. Questions: compute the P- upper and lower approximations, boundary, positive region and negative region on set X.

Answer:

The following information are obtained from Table 3.1,

$U = \{e_1, e_2, e_3, e_4, e_5, e_6\}$,
$A = \{Headaches, Muscularpains, Temperature, Influenza\}$,
$C = \{Headaches, Muscularpains, Temperature\}$,
$D = \{Influenza\}$,
$V_{Headaches}$=\{yes, no\},
$V_{Muscularpains}$=\{yes, no\},

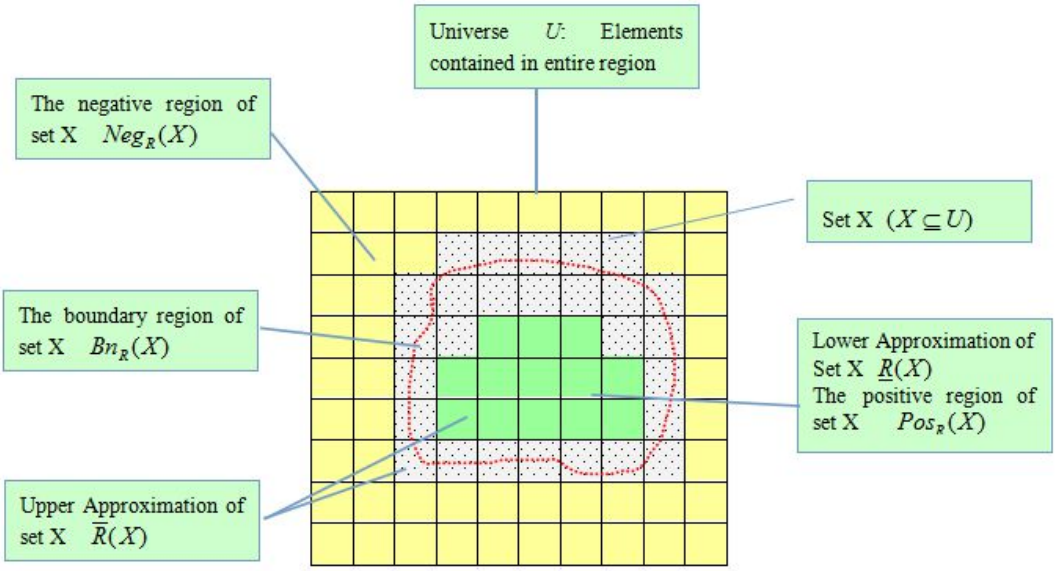


Figure 3.1: The lower and upper approximations of Set X

Table 3.1: A medical diagnosis decision system

Universe U	Condition attributes C			Decision attribute D
$Patients$	$Headaches$	$Muscularpains$	$Temperature$	$Influenza$
e_1	yes	yes	normal	no
e_2	yes	yes	high	yes
e_3	yes	yes	very high	yes
e_4	no	yes	normal	no
e_5	no	no	high	no
e_6	no	yes	very high	yes

$V_{Temperature}$={normal, high, very high},
$U/IND(Headaches) = \{\{e_1, e_2, e_3\}, \{e_4, e_5, e_6\}\}$,
$U/IND(Muscularpains) = \{\{e_1, e_2, e_3, e_4, e_6\}, \{e_5\}\}$,
$U/IND(Temperature) = \{\{e_1, e_4\}, \{e_2\}, \{e_5\}, \{e_3, e_6\}\}$,
$U/IND(P) = U/IND(Headaches, Muscularpains) =$
$U/IND(Headaches) \cap U/IND(Mucularpains) = \{\{e_1, e_2, e_3\}, \{e_4, e_6\}, \{e_5\}\}$.

The relations between set $X = \{e_2, e_3, e_5\}$ and basic set of P as below:
$X \cap \{e_1, e_2, e_3\} = \{e_2, e_3\} \neq \phi$;
$X \cap \{e_4, e_6\} = \phi$;
$X \cap \{e_5\} = \{e_5\} \neq \phi$;
P-lower approximation $\underline{R}(X) = \{e_5\}$;

P-upper approximation $\overline{R}(X) = \{e_1, e_2, e_3, e_5\}$;
P-boundary region $Bn_R(X) = \overline{R}(X) - \underline{R}(X) = \{e_1, e_2, e_3\}$;
P-positive region $Pos_R(X) = \underline{R} = \{e_5\}$;
P-negative region $Neg_R(X) = U - \overline{R}(X) = \{e_4, e_5\}$.

3.2.5 Knowledge reduction

Knowledge reduction is important in intelligent processing, it is one of the core content in rough set theory. On general, the attributes and equivalence relations in knowledge base are not equally important, even some knowledge is necessary or redundancy. Knowledge reduction means that maintain the ability of classification of the attributes set to delete the unnecessary knowledge.

Definition 5 Given a knowledge base $K = (U, S)$ and an equivalence relation cluster $P \subseteq S$, $\forall R \in P$, if

$$IND(P) = IND(P - \{R\})$$

then knowledge R is redundancy to P, else R is necessary to P. If every $R \in P$, R is necessary to P, then P is independent, else P is dependent to P.

Theorem 1 If knowledge P is independent, $\forall G \subseteq P$, then G is independent too.

Definition 6 (Knowledge reduction)

Give a knowledge base $K = (U, S)$ and an equivalence relation cluster $P \subseteq S$, for any $G \subseteq P$, if G satisfies the two conditions:

(1) G is independent;

(2) $IND(G) = IND(P)$.

then G is a reduction of knowledge P, it is donated by $G \in RED(P)$, whereby, $RED(P)$ represents the reduction set of P.

Definition 7 (Knowledge Core)

Given a knowledge base $K = (U, S)$ and an equivalence relation cluster $P \subseteq S$, for any $R \in P$, if R satisfies

$$IND(P - \{R\}) \neq IND(P)$$

then R is necessary to P, the set consisted in necessary knowledge to P called core of P, is donated by $CORE(P)$.

Theorem 2 $CORE = \cap RED(P)$

Theorem 2 demonstrates that knowledge core is the intersection of all the knowledge reductions, it means knowledge core is concluded in every knowledge reduction and can

be computed directly. In addition to this, knowledge core can't be reduced, if not, it will be weaken the ability of knowledge classification.

3.2.6 Rules extraction

Extracting rules from knowledge expression system is one of the main tasks in the field of data mining and knowledge discovery. Normally, four types of rules can be mined from data, such as characteristic, association, discriminant, and classification rules[5]. Rules induced from the lower approximation of the concept certainly describe the concept, hence such rules are called certain. On the other hand, rules induced from the upper approximation of the concept describe the concept possibly, so these rules are called possible.

3.3 Conclusion

In this chapter, we have presented the basic concepts of rough set theory. Our study based on rough set theory, so it is necessary to introduce the related knowledge of it. Rough set is a system theory. Rough set theory is a data mining tool to mine useful information from dataset. Once understanding the related knowledge of rough set theory, it becomes not difficult to know the latter works we done. In the next chapter, we will present the application of the rough set theory in cloud service selection

Chapter 4

Application of the rough set theory in cloud service selection

4.1 Introduction

Cloud computing has become a hot issue in the information technology society. It promises the ability to efficiently provide all types of services, which include utility computing, data storage, and software services available via the Internet to users with dynamic demands. In a pay-as-you-go manner, users consume computing resources to run their jobs and pay as much as the cloud providers charge them. For example, companies will purchase cloud service and get results as quickly as their programs can scale instead of investing in hardware deployments and human hiring.

With the rapid proliferation of cloud services providers, it is difficult for cloud users to know which ones are a good fit for their needs. Similarly, the cloud services providers need to improve their services to attract more cloud users. Here, we will give an approach to safeguard the interests of cloud users and cloud services providers.

For cloud service providers, the major challenge is exploiting the benefits of cloud computing to manage quality of service commitments to customers throughout the life cycle of a service. Users aim to get the cloud service at lowest price. There are lots of the cloud services with the same or similar functions but uneven quality. In addition, cloud service is a dynamic and open environment. Events often occur such as the increase or decrease dynamically of the cloud service, the service failure or variation. So users not only need to assess the quality of service but also balance the quality of service and outplay used to purchase cloud service to make the right choice. However, a variety of factors may influence the users' choice of the cloud service. Many users are concerned with such issues as reliability, availability, timeliness, while others may

care for the price, integrity. Therefore, they are often entangled in what kind of cloud services is more suitable for them. There is a need for a decision support tool to help cloud users choose the appropriate cloud service.

4.2 The selection of tool in studying cloud service selection

The effect of classification algorithm or decision-making approach usually is related to the characteristics of data set because that data set has null values, noise, sparse distribution etc, or because that their attribute values are different, some are continuous, some are discrete, or some are mixed. The classic classifiers are used successfully in many diverse areas. Such as decision tree classifier has been applied in medical diagnosticians, financial analyst, assess to credit risk of loan applicant etc; SVM (support vector machine) has been applied in pattern recognition, gene analysis, text classification, speech recognition, regression analysis etc; neural network classification algorithm is widely used in optical character recognition, molecular biology, face recognition etc because that it is not sensitive to noise data. As each classification algorithm or decision-making tool has its advantages and disadvantages, the diversity of the data and the complexity of practical problems, it is difficult to say which is better than other one. For example, neural networks is a learning algorithm based on the principle of empirical risk minimization, there exists some inherent shortcoming. However, SVM algorithm makes up them. So, in practical, choosing the right classification is key for specific problem.

Starting with the research on the satisfaction of cloud service users' demands, we take into consideration various factors, then we choose rough set theory as the research tool. The Rough set method is a well-known data mining technique having interesting advantages. In fact, rough set theory does not depend on any experience knowledge but it relies on data. It deals with the imprecise, uncertain or incomplete information without a priori of knowledge to induct the rules which is used to make the relevant decisions. It is not only able to assist providers to develop their service packages but also could help users to choose the cloud service with cost effective suited to their needs. Here, the first issue we are interested in concerns helping users to choose the cloud service using the rough set theory. This latter provides good properties for discovering and simplifying the factors involved in user choice.

In this chapter, we focus on the following problems that how cloud users can make

the decision among the cloud service providers and how cloud providers can obtain more customers. We propose a solution for extracting the important indicators of cloud service system based on rough set theory. We firstly determine the crucial factors to choose all kinds of the cloud services for users. We define cloud service items as a set of the objects, the factors as the attributes of these objects, the attribute values of the objects are the relevant data collected. Based on that, we establish the information system. Then, we use rough set theory to reduce the attributes and to mine the rules that will help users in making decisions about selecting suitable cloud service.

4.3 Related works

With highly developed information technology, it is obvious that cloud computing is becoming the future of enterprises and institutions. More and more cloud service providers have emerged, users need efficient and automated solution to select appropriate cloud service that fit their requirements. As cloud service selection is highly similar to web services selection and as very few works exist on automated and efficient selection of cloud service, we will give a brief introduction about some recent researches on web services. Literatures[127] [128] [129] focus on the optimal web service composition problem and proposed different algorithms to facilitate the delivery of high quality composite web services. In literature[130], the authors proposed a hybrid genetic algorithm with conflict constraint for the optimal web service selection problem from the computational point of view. In literature[131] presented a global quality of service optimizing and multi-objective Web services selection algorithm based on multi-objective ant colony optimization for the web service composition. In literature[132], an efficient service selection scheme in web services is proposed, which could help service requesters select different web services. However, cloud service is different from the traditional web service. The research objects are computing and software resources in traditional web service, but in cloud service pattern, except the above-mentioned two kinds of resource, it includes hardware resource, storage service and other features. In addition, the users' selection of the traditional web service focuses on the service indexes of the quality of service. As cloud provides a pay-on-demand service, users emphasize on price, status of service, response time and so on.

Zia et al. [133] proposed a user-feedback-based approach to monitor cloud performance, which rely on the data gathered from cloud users. However, it lacks of objective assessment that should allow users to get comprehensive performance of a cloud service through authors' framework. A cloud service algorithm is proposed in

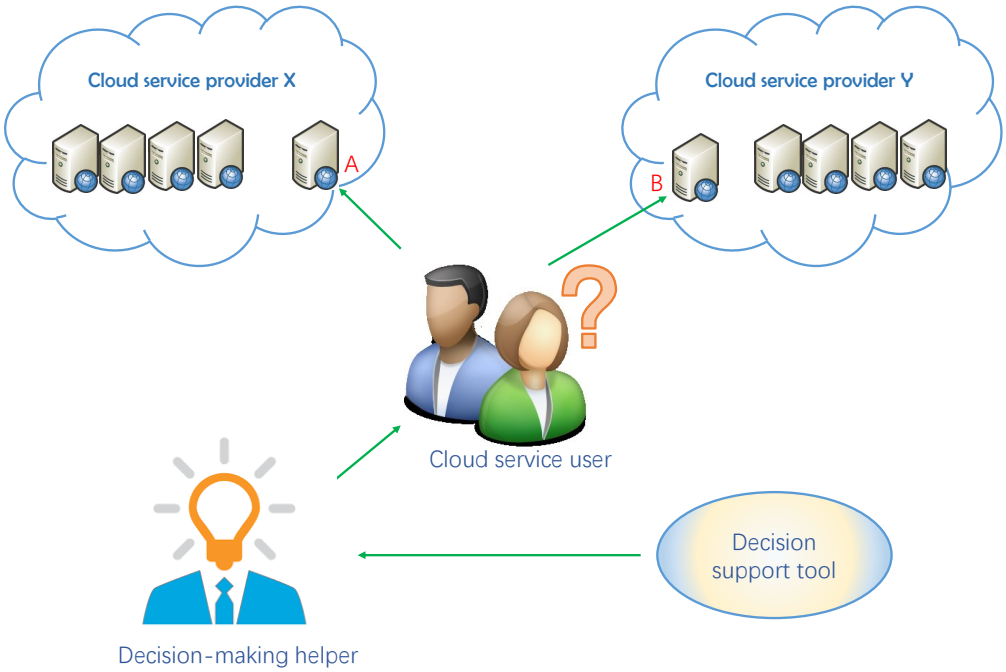


Figure 4.1: Cloud user decision helper

literature[134]. It discussed the cloud service architecture and gave an algorithm about service selection with adaptive performances and minimum cost. In literature[14], authors proposed a model of cloud service selection by aggregating the information from both users' feedback and objective performance analysis from a trusted third party. The proposed model is very similar to traditional web service and do not take into account the pay-on-demand feature of cloud system. In literature[15], authors formalize the cloud service selection problem into a rigorous mathematical form and presented a multi-criteria cloud service selection methodology using this formalism, which be used to service selection from among services with similar specific functions.

4.4 A framework of the rough set theory in cloud services

When there are many services in cloud, users hope quickly to select services from the corresponding candidate sets. In this part, we adopt rough set theory to build a cloud service selection model to help users make efficient decision for users. The main idea consists in calculating lower and upper approximations based on specific characteristic of attributes and then producing the rules for services selection.

As the figure 5.4 shows, when cloud service users need some cloud service, they

may encounter a confused situation where two different cloud service providers X and Y both provide the same kind of cloud service. According to the users preference, it is hard to tell out whether the service from provider X is better than that from provider Y. That is to say that one property of service provided by X may be better than that service provided by Y, while Y provide a better quality of service of another property for cloud users. Even though it seems clear that the overall quality of service of X is more suitable for cloud users, it is still difficult for the cloud users to decide directly to accept the service from which provider. Because higher quality of service usually means higher cost. Instead of making the choice by the cloud users themselves, a decision-making helper can choose the best service provider for the service requirement of cloud users. The core part of the decision-making helper is the decision support tool. It takes the cloud users' preferences and the properties of services from different providers gathered by decision-making helper as input. It can make the best choice for cloud users, which can help the mobile users choose the service effectively and accurately.

As the knowledge is generally not equally important, with unnecessary or redundant items, knowledge reduction concept is used. Knowledge reduction aims to maintain the classification ability of the knowledge base under the certain conditions of removing unnecessary knowledge. The process of reducing information leads to a set of attributes that are independent and no further can be deleted without losing consistency. The process of reducing knowledge information is also known as attributes reduction [4].

Extracting rules from knowledge expression system is one of the main tasks in the field of data mining and knowledge discovery. Normally, four types of rules can be mined from data, such as characteristic, association, discriminant, and classification rules [15]. Here, we focus on extracting the association rules from the information system we constructed. These rules will help users in making efficient selection of cloud service. The decision-making process of cloud service selection is illustrated in Figure 4.1.

Based on the work flow described in Figure 1, we construct the corresponding cloud service candidate sets and their attribute sets (the subjective and objective assessment metrics) to generate the information system.

Some trusted third parties and monitoring centers of cloud service analyze the performances of cloud service based on the data collected from cloud users' feedbacks. By combining cloud service characteristics, many metrics can be quantitatively measured (e.g., availability, elasticity, service response time, and cost per task). We can segment assessment metrics level, such as memory Reading/Writing, throughput, the speed of CPU and so on. As the company's data security and privacy are crucial, security and

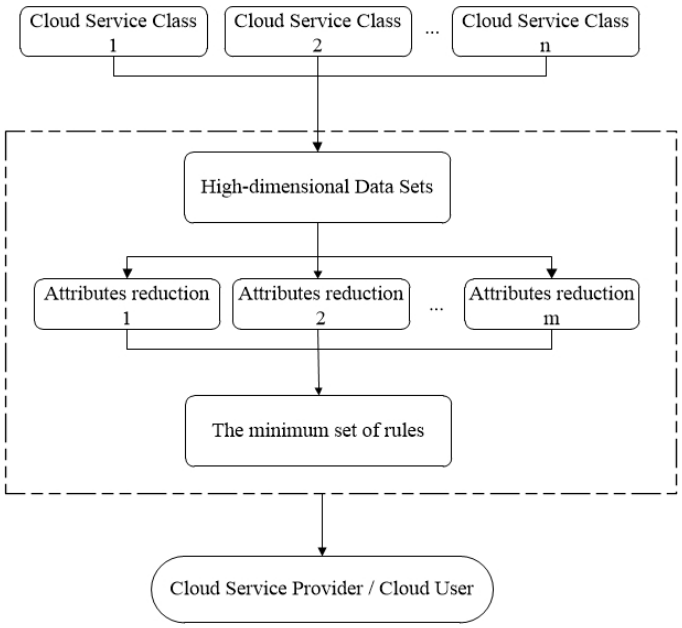


Figure 4.2: Cloud service selection based on rough set theory

privacy could also be the assessment criteria. The attribute values can be extracted from the magnanimity date sets.

The massive amounts of raw data usually make decision process very complicated. Since rough set methods deal only with discrete attributes, a series of pre-processing such as discretization of some continuous attributes is necessary.

The information system falls into two types: the complete and the incomplete information system. Incomplete information system is the one with missing values of some attributes. In reality, most of the information systems are incomplete. Recall that one of the biggest advantages of the rough set is that it can deal with imprecise, inconsistent and incomplete information, which motivate this work and the selection of this mining tool.

When dealing with incomplete information systems, there are two ways to achieve knowledge reduction: First consists in changing the incomplete information system into a complete one through data remove or complement. Second is to set null as default value for missing data. After pre-processing data, attributes are reduced and the minimum set of rules is deduced. In the following, we will give an example of cloud service selection based on rough set theory in which we apply knowledge reduction.

4.5 An example of classification and decision-making

In this section, we present the details of application of rough set theory in cloud service selection through a simple example.

4.5.1 Relevant definitions

The following are the relevant definitions about the process of attribute reduction and rules induction:

Definition 1 [116][118] The 4-tuple $DT = (U, C \cup D, V, f)$ is a decision information system, where $U = \{X_1, X_2, \ldots, X_n\}$ is a finite set of objects and $|U| = n$. We define the discernibility matrix of the decision information system as follow,

$$M_{n\times n}(DT) = (c_{ij})_{n\times n} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{bmatrix}$$

where $i, j = 1, 2, \cdots, n$.

$$c_{ij} = \begin{cases} \{\alpha | (\alpha \in C) \wedge (f_\alpha(x_i) \neq f_\alpha(x_j))\}, \\ \qquad\qquad f_D(x_i) \neq f_D(x_j); \\ \emptyset, \\ f_D(x_i) \neq f_D(x_j) \wedge f_C(x_i) \neq f_C(x_j); \\ -, \qquad\qquad f_D(x_i) = f_D(x_j). \end{cases}$$

c_{ij} is the element in discernibility matrix.

According to definition 2, information function $f_\alpha(x_i)$ denotes a value for the condition attribute α of the object x_i. Information function $f_D(x_i)$ denotes a value for the decision attribute D of the object x_i.

Definition 2 Let 4-tuple $DT = (U, C \cup D, V, f)$ be a decision information system, where $U = \{X_1, X_2, \ldots, X_n\}$ is a finite set of objects and $|U| = n$. $\forall \alpha \in A, \forall X_i, X_j \in U$, we order the discernibility variable with respect to attribute α as follows:

$$\alpha(X_i, X_j) = \begin{cases} \{\alpha | (\alpha \in C) \wedge (f_\alpha(x_i) \neq f_\alpha(x_j))\}, \\ \qquad\qquad f_D(x_i) \neq f_D(x_j); \\ \emptyset, \\ f_D(x_i) \neq f_D(x_j) \wedge f_C(x_i) \neq f_C(x_j); \\ -, \qquad\qquad f_D(x_i) = f_D(x_j). \end{cases}$$

It equals the element c_{ij} in discernibility matrix. So, we have

$$\Sigma\alpha(x_i, x_j) = \begin{cases} \alpha_{l_1} \vee \alpha_{l_2} \vee \cdots \vee \alpha_{l_k}, \\ \{\alpha(x_i, x_j) = \alpha_{l_1}, \alpha_{l_2}, \cdots \alpha_{l_k}\} \\ \qquad\qquad (1 \leq k \leq card(c); \\ -, \qquad \alpha(x_i, x_j) = \emptyset \vee -. \end{cases}$$

The discernibility function is then defined as follow:

$$\Delta = \prod_{\forall (x_i, x_j) \in U \times U} \sum \alpha(x_i, x_j) \stackrel{\text{def}}{=} \bigwedge_{\forall (x_i, x_j) \in U \times U} \sum \alpha(x_i, x_j),$$
$$i, j = 1, 2, \cdots, n.$$

The discernibility matrix and discernibility function are used to reduce redundant knowledge.

Definition 3 [116][118] Let 4-tuple $DT = (U, C \cup D, V, f)$ be a decision information system. Let C,$D \subseteq A$. Obviously if $C^{'} \subseteq C$ is a *D-reduct* of C, then $C^{'}$ is a minimal subset of C. We will say that attribute $\alpha \in C$, if $Pos_C(D) = Pos_{(C-\{\alpha\})}(D)$, then subset $C^{'} = (C - \{\alpha\}) \subseteq C$ is a *D-reduct of* C denoted as $RED_D(C)$. $CORE_D(C) = \bigcap RED_D(C)$ will be called *D-core* of C.

4.5.2 Application of rough set theory to sample dataset

According to the part of the analysis about the assessment index of the cloud service in section 4.3, we established a simple instance given in Table 4.1. Without losing generality, we assume a complete information system, and we choose some keywords as the attributes. Then, all the attribute values are processed with the discretization.

Table 1 represents the decision information system. $U = \{X_1, X_2, \ldots, X_{14}\}$ is the universe that corresponds to the cloud service set. $C = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ is the set of

condition attributes, where α_1, α_2, α_3 and α_4 are respectively the response speed, the service feedback, the price per task and the rapid elasticity. $D = \{d\}$ is the decision attribute, where (d) is the cost effectiveness.

It is easy to notice how much it is complicated with such a data set to make an efficient decision on the cloud service selection, without the use of any further tool. Also, the amount of available data is pretty much higher than a table in 14 rows and 5 columns.

Table 4.1: The decision information system of the cloud service selection

Universe	Condition Attribute				Decision Attribute
U	α_1	α_2	α_3	α_4	d
x_1	fast	bad	high	yes	low
x_2	fast	bad	high	no	low
x_3	normal	bad	high	yes	high
x_4	slow	good	high	yes	high
x_5	slow	very good	normal	yes	high
x_6	slow	very good	normal	no	low
x_7	normal	very good	normal	no	high
x_8	fast	good	high	yes	low
x_9	fast	very good	normal	yes	high
x_{10}	slow	good	normal	yes	high
x_{11}	fast	good	normal	no	high
x_{12}	normal	good	high	no	high
x_{13}	normal	bad	normal	yes	high
x_{14}	slow	good	high	no	low

The detailed procedure of our approach is shown below:

Step 1 (discernibility matrix)

By using reduction method, all objects are discernible in the information system. According to definition 3, the obtained discernibility matrix from Table 4.1 is :

The obtained discernibility matrix is :

$$M_{14\times14}(DT) = (c_{ij})_{14\times14}$$

$$= \begin{bmatrix} - & & & & \\ - & - & & & \\ \vdots & \vdots & \ddots & & \\ \{\alpha_1,\alpha_3\} & \{\alpha_1,\alpha_3,\alpha_4\} & \cdots & - & \\ - & - & \cdots & \{\alpha_1,\alpha_2,\alpha_3,\alpha_4\} & - \end{bmatrix}_{14\times14}$$

Step 2 (Attributes Reduction)

According to definition 4, we reduce redundant knowledge which is invalid for making decision in Table 4.1 as below:

The 45 disjunctive logic expressions which meet "non empty" and "non -" are extracted from the discernibility matrix. We get:

$$\begin{aligned} L_{1,3} &= \alpha_1, \\ L_{2,3} &= \alpha_1 \vee \alpha_4, \\ L_{1,4} &= \alpha_1 \vee \alpha_2, \\ L_{2,4} &= \alpha_1 \vee \alpha_2 \vee \alpha_4, \\ &\vdots \\ L_{13,14} &= \alpha_1 \vee \alpha_2 \vee \alpha_3 \vee \alpha_4 \end{aligned}$$

After performing logical conjunction on those expressions we obtain the following conjunctive logic expression:

$$\begin{aligned} L_{\wedge}(\vee) &= L_{1,3} \wedge L_{2,3} \wedge L_{1,4} \wedge \cdots \wedge L_{13,14} \\ &= \alpha_1 \wedge (\alpha_1 \vee \alpha_4) \wedge (\alpha_1 \vee \alpha_2) \wedge \\ &\quad (\alpha_1 \vee \alpha_2 \vee \alpha_4) \wedge \cdots \wedge (\alpha_1 \vee \alpha_2 \vee \alpha_3 \vee \alpha_4) \end{aligned}$$

Transforming $L_{\wedge}(\vee)$ give the conjunctive form:

$$L'_{\vee}(\wedge) = (\alpha_1 \wedge \alpha_2 \wedge \alpha_4) \vee (\alpha_1 \wedge \alpha_3 \wedge \alpha_4)$$

Step 3 (Core of the attributes)

According to definition 5, the $RED_D(C)$ set contains all the relative attributes reduction of the decision information system regarding the decision attribute and is given by:

$$RED_D(C) = \{\{\alpha_1,\alpha_2,\alpha_4\},\{\alpha_1,\alpha_3,\alpha_4\}\}$$

When calculating $Pos_{C-\{\alpha_2\}}(D)$ and $Pos_{C-\{\alpha_3\}}(D)$ we notice that it is equal to $Pos_C(D)$. Thus, the condition attribute α_2 or α_3 is unnecessary for decision attribute D. Thus, condition attributes α_1 and α_4 are then the core of the reduction attributes.

$$CORE_D(C) = \{\alpha_1, \alpha_4\}$$

Core is the common attributes which are in reductions sets. In other words, condition attributes α_1 and α_4 are necessary, they can never be reduced from information table. Deleting any of them will affect the classification ability with equivalence relation.

Step 4 (Generated rules)

According to the two above attributes reduction results, we randomly select one of them to generate the associate rules such as the attribute reduction α_1, α_2, α_4. Based on the definitions 1 to 4, the some decision rules are the following:

$$R1\ (\alpha_1, fast) \wedge (\alpha_2, bad) \wedge (\alpha_4, yes) \rightarrow (d, low)$$
$$R2\ (\alpha_1, fast) \wedge (\alpha_2, bad) \wedge (\alpha_4, no) \rightarrow (d, low)$$
$$R3\ (\alpha_1, general) \wedge (\alpha_2, bad) \wedge (\alpha_4, yes) \rightarrow (d, high)$$
$$R4\ (\alpha_1, general) \wedge (\alpha_2, verygood) \wedge (\alpha_4, no) \rightarrow (d, high)$$
$$R5\ (\alpha_1, general) \wedge (\alpha_2, good) \wedge (\alpha_4, no) \rightarrow (d, high)$$
$$R6\ (\alpha_1, low) \wedge (\alpha_2, good) \wedge (\alpha_4, yes) \rightarrow (d, high)$$
$$R7\ (\alpha_1, low) \wedge (\alpha_2, verygood) \wedge (\alpha_4, yes) \rightarrow (d, high)$$

As decision system contains a lot of information samples, each sample forms a basic decision rule, so there may be a lot of redundant rules. To obtain minimal decision rules to guarantee the ease of use which our main goal, we will reduce the basic set of rules.

For decision rules with same decision values, if there are condition attributes with different values, then it is possible to reduce these attribute values to obtain the minimum rule set. For example, in decision rules *R1* and *R2*, the decision attribute d with the same value *low*, and the values of the condition attribute α_4 are different, so we can reduce these two rules. Hence, *R1* and *R2* are combined into rule $R_1^{'}$. Similarly, *R3*, *R4* and *R5* are combined into rule $R_2^{'}$ and so on. In the following are given the minimum set of rules we obtain after reduction:

$$\begin{aligned} R_1^{'} &= (\alpha_1, fast) \wedge (\alpha_2, bad) \rightarrow (d, low) \\ R_2^{'} &= (\alpha_1, general) \rightarrow (d, high) \\ R_3^{'} &= (\alpha_1, low) \wedge (\alpha_4, yes) \rightarrow (d, high) \end{aligned}$$

Analysis and interpretation of the results decision rules as follows:

Rule $R_1^{'}$: Even if response speed of the cloud service is fast, but the user feedback is bad, this leads to the cost effectiveness is low.

Rule $R_2^{'}$: Only if the response speed value of the cloud service is general, however, the cost effectiveness is high. When users choosing the cloud service, the values of the other indexes of the cloud service can be ignored.

Rule $R_3^{'}$: Cloud service has high cost effectiveness when it is valid of the rapid elasticity, although response speed is low.

These three rules give meaningful information for the cloud users and the cloud service providers. Cloud users can rely on these rules to make efficient decision. And cloud service providers can improve the quality of the cloud service focusing on particular aspects according to these decision rules.

The reduction algorithm of discernibility matrix is described as follows:

Algorithm 1 Attribute reduction algorithm of discernibility matrix DM

Input:

The information system of cloud services;

Output:

The attributes Reduction of the cloud services system: Red;

1: Input the information table of cloud services;
2: set Red=ϕ, count(a_i)=0, for i=1, n;
3: compute the discernibility matrix and weight frequent of attributes count(a_i); \\ every new item C of DM, count(a_i):=count(a_i)+n/ $\mid c \mid$, $a_i \in \mid c \mid$.
4: merge all the same items and order the discernibility matrix according to the length of item and frequent;
5: for each m of DM ;
6: if ($m \bigcap Red == \phi$);
7: choose the attribute a of m, maxi=count(a);
8: Red=$Red \bigcup \{a\}$
9: end if;
10: end for;
11: return Red.

We test the algorithm with Java. It is executed on a processor Inter Core 2 Duo CPUs x64. we firstly test the example 1, the result shows that our method is valid. Secondly, we adopt data sets (download from the UCI [27]) to run the algorithm, we get the good results also.

4.6 Conclusion

Rough set theory is a useful tool for analyzing big datasets, which can be used to mining the information hidden in datasets. In this chapter we proposed a cloud service selection model based on the rough set theory to help cloud users making efficient decision. On a simple example and given some key assessment attributes according to the objective and subjective metrics, we had reduced the redundant knowledge and we deduce the associate rules. Those were also reduced to get the minimal set in order to propose easy and efficient selection system. In the next chapter, we will introduce the evaluation method for the parameters importance of cloud service selection using rough set theory.

Chapter 5

Evaluation of parameters importance in cloud service selection using rough set theory

5.1 Introduction

For several years, cloud computing has been influencing the IT landscape and becomes an important economic factor [96] due to its mode of operation that is the pay-as-you-go to provide service. Since cloud computing is a minimal barrier to entry and economic scaling, there are a lot of prospective clients to move their business on it. In this context, many small and large cloud service providers emerge every day. However, not all of them are the first-hand owners of a cloud infrastructure. This means that for those smaller cloud service providers, they are only partnered with a bigger provider which owns the infrastructure. Normally this is not a big problem, even though they are all connected to a bigger infrastructure provider, when it goes down, all "middle-man" go down with it. Since cloud service providers have their specific service model, therefore, it is difficult for users to compare the cloud services offered by the different providers. Consequently, the cloud user faces a challenge to select an appropriate provider taking into account his specific requirements.

Some cloud users take into consideration their subjective preference parameters of the assessment criteria, while ignoring the importance of objective assessment parameters obtained from other customers who had the same service requirements when they are selecting the cloud services. Most cloud users can not find an appropriate cloud service matching their individual requirements when they are using a given cloud service for the first. In fact, as they are not sure that the performance and quality of

the selected service are good, they choose on the basis of their subjective judgment to the adapted decision parameters. Furthermore, when cloud users try to give an overall assessment for a cloud service, it is also not objective that the parameter weights of cloud service are generated by usually subjective experience or experts scoring. This affects the cloud users choice of a suitable cloud service.

For all the issues mentioned above, we can obtain the importance rating of attributes and rank them through the rough set theory, thereby we determine the objective weight of the assessment indexes of cloud services. Our proposal not only can guide cloud users, facing a lot of choices of cloud services, concerning assessment indexes they should focus, but also helps cloud providers to improve the performance and quality of the cloud services with the emphasis to attract more cloud users to make themselves have a predominance in future competition of IT industry.

5.2 Related works

With the development of cloud computing technology, the cloud service is becoming a mature concept concerning the delivery of software services, infrastructure services and platform services. Many techniques have been proposed by researchers from academia and industry for cloud services publication, interface definition and service discovery. Cloud service techniques(e.g., virtualization technique) have greatly accelerated the adoption and deployment of cloud services.

At the same time, more and more cloud service providers are offering all kinds of cloud services. For users, it is difficult to make decision about the services meeting their requirements. To allow customers to evaluate cloud offerings and rank them based on their ability to meet the user's QoS (Quality of Service) requirements, Garg,S.K. et al. proposed a framework and a mechanism that evaluate the quality and rank cloud services [96]. In this framework, the authors presented a rank cloud services mechanism using AHP (Analytic Hierarchy Process) [97] for solving problems related to MCMD (Multiple-criteria Decision-making). AHP is a widespread service ranking method. It is a structured technique for organizing the cloud service information and analyzing complex decisions. The analytic network process (ANP) [98] can provide a solution to problems that cannot be structured hierarchically, and is considered as an extension of AHP. An AHP-based SaaS services selection method is introduced in literature [99] to score and rank services. The researchers construct an AHP hierarchy to represent SaaS service attributes. Although the use of AHP can improve the objective rating based on selection attributes, however, the importance of the service attributes is judged by

aggregating user preferences and the opinions of experts, so the result of services ranking is more subjective. On the basis of AHP hierarchy, N. Boussoualim [100] proposed an approach to calculate the weights of the various attributes of choice parameters and score the different products in an SaaS selection to help users to make decision. Since weights of various factors are assigned according to the user preferences, therefore, this method is also limited by the subjective judgment. Karim et al. [101] defined an AHP hierarchy of a cloud service weighting model, in which a mechanism (a set of rules to perform the mapping process) is explored to map the users' QoS requirements of cloud services to the right QoS specifications of SaaS. Nie G.h. et al. [102] proposed a cloud service evaluation index system to guide users in the choice of cloud services. These works have some common features, such as the proposed models are based on AHP, the initial importance of the parameters based on subjective judgment and so on.

Unlike AHP, other approaches for cloud service selection are proposed. Han S.M. et al. [103] presented a cloud service selection framework in the cloud market to help users select the better services. This cloud service recommendation system is based on a utility function to quantify the preferences of a decision maker. In [104], authors described a framework for reputation-aware software service selection and rating. It aims to rate SaaS services while reducing the time and risk of the selection and utilization of software services. The proposed selection mechanism aids service users to select services based on quality, cost and reputation. Saripalli et al. [105] discussed Multiple Attribute Decision Methodology to rank alternatives in a decision problem in cloud service adoption. In this work, the authors analyzed the possible decision problems the service users might encounter. The Simple Additive Weighting (SAW) method is used to rank the service candidates based on the rating values generated.

In mentioned above works, the researchers proposed various ranking approaches for cloud services selection. To rank the cloud services, it is necessary to evaluate the importance of the parameters given in cloud services selection. Since the weight for each parameter acquired by conducting experts opinions or user preferences in above works, as a result, certain recommended cloud services are not always the best to meet users' requirements.

Different from research emphasis of the above works, our study focuses on the parameters importance evaluation to guide users in cloud services selection. To get a rational evaluation result for each cloud service parameter, we use the rough set theory to carry out our work. In [106], the author proposed an approach for mining significant factors affecting the adoption of SaaS using the rough set theory. Although we are using the same theory in a similar context, our work makes a further study. The method we

proposed not only can explore the significant factors but also can rank and weight these parameters in cloud services selection.

5.3 Evaluation Parameters of Cloud service

With the rapid development of cloud computing, more and more cloud service providers join cloud market. Businesses and consumers have more choices because a large number of industry application solutions emerge. The global market scale for cloud services is increasing. Cloud computing providers carry on the business on a unified platform by building cloud resource pool for resource sharing, resource centralization, service network, billing and demand elasticity, to achieve cloud business structure on a scale. From a marketing perspective, the main types of cloud services are cloud hosting services, object storage services, cloud database services, cloud engine services, block storage services, cloud caching services, online application services, load balancing services and cloud distribution services. From another perspective, cloud services include IaaS, Paas and SaaS. Moreover, cloud can be divided into public cloud, private cloud and hybrid cloud on deployment.

The core business is various from different cloud service providers. For example, Amazon's business is more interested in the platform and software (PaaS and SaaS), which are public cloud services. However, IBM has a wider range for business, and its hardware and platforms are more advanced; IaaS, PaaS, SaaS and other aspects of the business are involved, more favored in building private and hybrid clouds. Therefore, it is difficult for the user to define what cloud service providers are the best on the basis of a certain point. There are some configuration parameters for every type of cloud services to evaluate their performance. For example, the number of CPU, the size of memory, the space of storage, operate system and so on, these parameters determine the performance of cloud hosting services. When users are choosing one type cloud service, there are many alternative cloud service providers. When the users make choices, they need some parameters to evaluate cloud service providers' comprehensive ability, such as the capacity for innovation, the service capability, product technologies, the solutions, brand influence, etc. Usual evaluation parameters of cloud service and cloud services providers as follows:

- Cloud service availability

  Availability is the proportion of time a system in functioning condition. For cloud service availability, it can be defined as the capacity of an IT system to

provide continuous service delivery. We give an example to understand what exactly it means. Let's take a 99.9% SLA, in practice, this means that in any given month (assuming a 30-day month), the service can only be unavailable for about 4 minutes and a few seconds, or only about 50 minutes per year. It includes connectivity, reliability, delay, data leakage and loss, cyber attacks, and the tenant's business does not meet expectations or entirely suspended caused by any accident on IaaS, PaaS and SaaS. As cloud services mature, cloud service availability becomes as important as price or other factors in choosing the right service provider.

- Cloud service scalability

  Scalability is a broad concept. It appears in a wide range of applications. For cloud service, scalability is the ability of the whole system to sustain increasing workloads by making use of additional resources. It is about how to deal with the large-scale business and attract more users. It is not directly related to how well the actual resource demands are matched by the provisioned resources at any point in time, even if there is more than a single point of failure. However, scalability of cloud service composition needs to meet the requirement for expanding users and technology upgrade.

- Cloud service elasticity

  Elasticity has become a key metric of cloud service. Elasticity is used in the naming of specific cloud products or service. It is an ability of a system to adapt to change in workloads and resource demands. Users expect to obtain the best service with the cheapest way. As we all know, cloud services provide multi-service contracts depending on the different hierarchical levels of users' needs. This dynamic proposition allows the users selecting the suitable options according to their needs and the amount of the resource they used. Therefore, users use the service quite flexibly with defined rights at any moment to save money. Usually, the term elasticity is one of the keywords for promoting the development of cloud service[109].

- Cloud service security

  Cloud service concerns a number of security issues[108]. such as software platform security and Infrastructure security via the cloud. Cloud service providers must ensure their clients' data and applications are protected, while users can through authentication enhance their application security. Cloud service providers often

store many users' data on the same server to save costs, conserve resources and maintain efficiency. As a result, there is a chance that user's private data can be viewed by other users without taking effective measures. Moreover, the precautionary measures to prevent Internet from hacking and virus damage. Therefore, cloud service security is an important index when evaluating the quality of the service.

- Capacity of innovation

  Innovation is described in terms of changes in what a company offers the product or service upgrade and the ways it creates and delivers those offerings (process improvement)[109]. Innovation is the soul of enterprise progress, the core of economic competition. An enterprise's ability to innovate is a key to its success. When most competitors within an industry have acquired the same level of competence in areas of management, such as marketing operations, human resources and strategy, they need to look for some innovations, such as incentive, resource investment and enterprise's self-fulfillment as a key factor for significant competitive advantages.

- Total Cost of Ownership

  Total Cost of Ownership (TCO) is an analysis technology to uncover all the lifetime costs that follow from owning certain kinds of assets. TCO provides a cost basis for determining the total economic value of an investment when incorporated in any financial benefit analysis[16]. TCO analysis attempts to uncover both the obvious costs and the "hidden" costs of ownership. Obvious costs in TCO are the costs involved during planning and vendor selection, such as purchase cost and the actual price paid. "Hidden" costs include acquisition costs, upgrade costs, security costs and so on. TCO is a scientific, rational economic evaluation index for firms.

- Service capability

  Service capability is the degree of capability in a service system to provide services and is commonly defined as the maximum output rate of the system. Compared with the manufacturing industry, service capability of IT enterprises stress the technology and skills to meet the needs of customers with high quality serving products[111][112]. Enhancing the service capability can improve the competitive advantages.

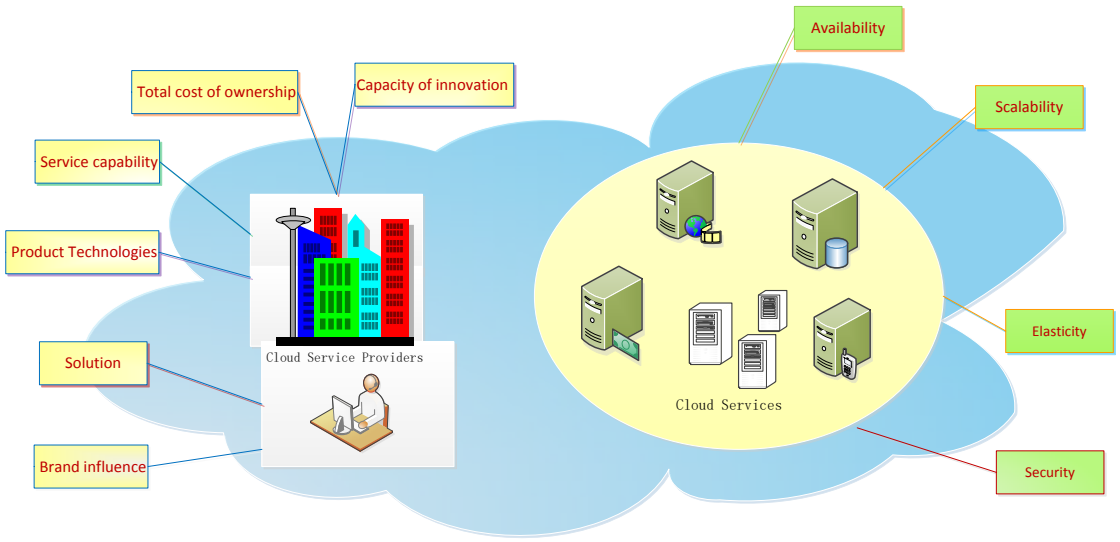


Figure 5.1: Evaluation parameters of cloud services and providers

- Solution

  For some problems (such as deficiency, demands, shortage) that already occurred or can be predicted in an enterprise, solution is a specific plan or proposal that can be effectively implemented. An excellent solution offers a series of conclusion: Why it happens? Whether it occurs again or not? Does it lead to other problems? How to avoid related problems? What experiences are accumulated from the solution? [120] As well as in some fields, solution should meet customers demands to achieve the expected effects.

- Brand influence

  Brand influence refers to the ability of opening up market and gaining the benefits with the brand[121][122]. It has been an important element for customers to choose their cloud service providers.

The evaluation parameters of cloud service and provider are shown in Figure 1.

5.4 Rough set theory

Rough set theory proposed by Pawlark in [116] is a mathematical approach to uncertain knowledge. Rough set theory has been applied in many interesting areas. The rough set approach is of fundamental importance to artificial intelligence and cognitive sciences, especially in the fields of machine learning, knowledge acquisition, knowledge discovery,

decision analysis, expert systems, inductive reasoning and pattern recognition[117]. The main advantage of rough set theory in the process of knowledge analysis is based on dataset rather than subjective judgement.

Definition 1 [117][118][119] Let $T = (U, A, V, f)$ be an information system, where $U = \{X_1, X_2, \ldots, X_n\}$ is the finite set of objects; $A = C \cup D$ is the set of attributes, C is a conditional attributes set, D is the decision attribute set; $V = \cup V_\alpha$, where V_α is the set of values of attributes $\alpha \in A$. f is an information function and denotes the map of $U \times A \longrightarrow V$, which assigns a value to each attribute for each object.

Definition 2 [117][118][119] Given an information system $T = (U, A, V, f)$, $A = C \cup D$. The expression $Pos_C(D)$, called a positive region of the partition U/D with respect to condition attributes C, is the set of all elements of U that can be uniquely classified to blocks of the partition U/D, by means of C. U/D indicates elementary concepts of information system T about decision attribute set D. For $\alpha \in C$, we have:

a If $Pos_{C-\{\alpha\}}(D) = Pos_C(D)$, then α is an unnecessary attribute of C;

b If $Pos_{C-\{\alpha\}}(D) \neq Pos_C(D)$, then α is a necessary attribute of C.

Definition 3 [118][119] Given an information system $T = (U, A, V, f)$, $A = C \cup D$. Attribute importance of the decision information system can be tested by the classification ability for T when removing an attribute $\alpha \in C$ from condition attribute set C, the significance of the attribute α is defined by [22] as:

$$Sig(\alpha) = \frac{|card(Pos_C(D))| - |card(Pos_{C-\{\alpha\}}(D))}{|U|}$$

Card presents the set cardinality of the attributes. $Sig(\alpha)$ represents the dependence of decision attribute D relative to condition attribute α, and which reflects the classification discrimination ability of the attribute α. The larger value of $Sig(\alpha)$, the more stronger of dependency relationships between condition attribute α and decision attribute D, and the more discriminative the attribute α is.

5.5 The cloud service selection method with preference information

Cloud users usually give the subjective weight to different parameters of the cloud service based on personal preference when they are choosing the cloud service, thus resulting into a non practical choice. Therefore, in this section we introduce an approach

to rank the importance of the cloud service indexes and provide the objective weight about different parameters based on the rough set theory.

5.5.1 The objective ranking of attributes approach based on rough set theory

Rough set theory analysis is based on upper and lower approximations space. The lower approximation of the set can describe the precise knowledge in an information system, which is called positive region and is defined by definition 2. If the lower approximation will not be changed when an attribute is deleted, then the attribute is unnecessary and can be reduced. Otherwise, the attribute is called core attribute, which is necessary. In other words, the definition 2 can distinguish the core attributes and unnecessary attributes while ignoring the effect of the relatively necessary attributes. For all relatively necessary attributes, we can rank them in an information system according to the significance values of different attributes. The significance of an attribute defined by definition 3 can reflect the variety of the lower approximation space when the attribute is deleted.

Since cloud service is characterized by various parameters, such as availability or scalability, elasticity and so on, it is difficult to define selection criteria valid for different customer needs. For this problem, we give a cloud service selection method using rough set theory, which is shown in the following:

We get the users' subjective preferences information through interacting with users. If some users provide incomplete information, we can adopt data complete mode translating the incomplete information into complete one. The method of getting user preferences information is shown in Figure 2.

First, we obtain the preference values of parameters of cloud services. Then, we compute the preference weight of various parameters. The user preference levels are shown in Table 1. To facilitate computations and storage in the database, we assign the preference levels with numerical values. * means that users do not provide personal preferences, which are null.

Table 5.1: The preference levels of users

Very important	Important	Not important	No selection
2	1	0	*

We construct an information system based on a large preference datasets collected from users of certain cloud service providers (google, Alibaba et al). Table 2 is an assess-

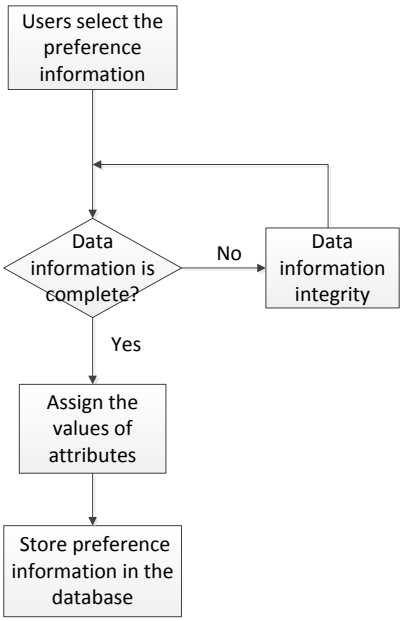


Figure 5.2: Getting the preference information

ment and requirement system of users about the cloud services. U represents the cloud services set, $U = \{s_1, s_2, \ldots, s_m\}$; Condition attributes set represents the assessment parameters of cloud services, $C = \{avalilability, scalability, reliability, credit, \ldots, loads\}$, that is $C = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$; decision attribute set is satisfied with the cloud service or not, $D = \{Yes, No\}$, that is, $\{1, 0\}$, where, * represents incomplete information.

Table 5.2: User preferences and assessment for cloud service

	α_1	α_2	α_3	α_4	...	α_n	d
s_1	1	*	2	1	...	0	0
s_2	0	0	1	1	...	1	0
s_3	1	2	1	2	...	*	1
s_4	2	1	0	0	...	1	1
s_5	1	2	0	*	...	0	0
s_6	0	0	2	0	...	0	1
s_7	1	*	1	1	...	1	1
...	...	...	...	...	...	...	0
s_m	*	1	0	2	*	1	1

To obtain the parameters importance of cloud service, the ranking of attributes algorithm is described as follow:

Algorithm 2 The ranking attributes of cloud services

Input:

The information system of cloud services;

Output:

The attributes ranking of the cloud services;

1: Input the information table of cloud services;

2: Set $C = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$;

3: Compute all partition U/D with respect to condition attributes C;

4: Set i=1;

5: if $i \leq$ *number of the attributes*;
then
Compute all partition U/D with respect to condition attributes $c = \{C - \alpha_i\}$;
i++;

6: Compute all the significance of the condition attributes with respect to decision attribute D

$$Sig(\alpha) = \frac{|card(Pos_C(D))| - |card(Pos_{C-\{\alpha\}}(D))}{|U|}$$

7: Rank the attributes of cloud services.

5.5.2 Application of the objective ranking of attributes approach in cloud service selection

Choosing the cloud services is a multiple attributes decision making problem, and the key is to determine the weight of parameters. There are several ways to determine the weight of indicators, on general, which fall into two categories: subjective and objective assignment methods. The subjective assignment method is assigning weight based on subjective information of decision-making. It is arbitrary with poor accuracy and reliability of decision-making. In the objective assignment method, each parameter is evaluated with the actual data. In cloud service selection system, the importance of attributes is different. The objective weight of attributes can be defined as:

$$W_\alpha = \frac{Sig_\alpha(\alpha)}{\sum_{c \in C} Sig_c(c)} \tag{5.1}$$

The comprehensive weight with regard to parameters can be defined as:

$$I(w) = \beta W_o(w) + (1 - \beta) W_{so}(w), 0 \leq \beta \leq 1 \tag{5.2}$$

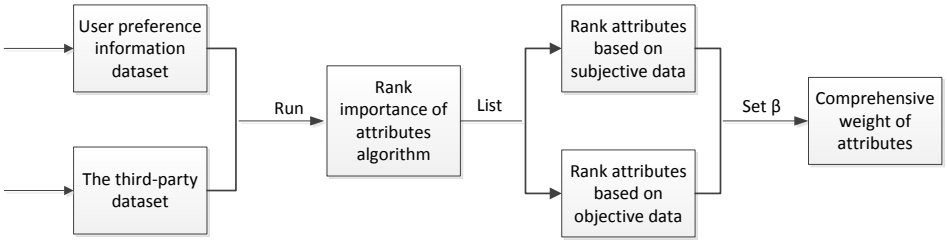


Figure 5.3: Application model of the objective ranking of attributes

Where, β which is called weight coefficient reflects cloud users preference for subjective and objective weights of parameters when they make decisions in cloud services selection. $W_o(w)$ and $W_{so}(w)$ respectively represents the weight of parameters of cloud services with objective dataset and subjective dataset. Smaller value of β indicates that users value more their subjective preference. Conversely, higher value of β users emphasizes the objective importance of parameters. Specially, if $\beta = 0$, users judging the parameters importance of cloud services totally depend on their subjective awareness; if $\beta = 1$, users completely rely on the objective weight.

An application is illustrated in determining the comprehensive weights of cloud service parameters based on the rough set theory. Obtaining the comprehensive weight of each parameter includes two parts. The first part is acquiring the weight of the parameters based on the subjective data which comes from the cloud user preferences. The second part is acquiring the objective weight based on the data without subjective information of decision-maker. The application model of the objective ranking of attributes in cloud service selection system is shown in Figure 3.

5.5.3 Application of attributes ranking approach in cloud service selection

There are corresponding indexes designed to evaluate a system or a service. When cloud service providers launch a service product to consumers, they should provide quality of services and they hope to get the feedback from consumers early to improve their products, at the same time, the evaluation indexes of the services to be design accordingly. For cloud service users, when they choose a cloud service, they will consider some factors to obtain the suitable service, such as cloud service availability, cloud service elasticity, brand of service etc. As we know, in economical market, the cost control and the pursuit of efficiency are the primary goals of each company management. The reason cloud users choose moving their business to cloud computing center is

because this is a good way to save capital and improve efficiency compare to their traditional development model. However, in practice, cloud users should balance the weight of factors used to evaluate cloud service.

Here we demonstrate an instance to use rough set theory to rank the factors of cloud service providers because the overall strength of cloud service provider is important for cloud users to choose the suitable cloud service. The real data in Table 3 is the list of cloud service providers according to their all-round capacity in 2014. The cloud service providers operate in China. The data is published in the journal of China Internet Weekly[26]. In Table 3, the factors CI (capacity for innovation), SC (service capability), PT (product technologies), S (solution), TCO (total cost of ownership) and BI (Brand influence) are the evaluation factors of cloud service providers. The factor CS (comprehensive score) is the assessment result of the cloud service providers.

In rough set theory, every cloud service provider is represented as a research object, and the factors as its attributes. Among them, the factor CS is decision attribute, while others are condition attributes. Simply, columns of Table 3 are labeled by attributes and rows - by objects, whereas entries of the table are attribute values. Thus, each row of the table can be seen as an information about specific cloud service provider. Our research purpose is to rank the weight of the factors to assess the comprehensive strength of cloud service providers.

We abstract randomly a cloud service provider from Table 3 to explain what it is the purpose we study, for example, *Amazon*. We can see from Table 3 that cloud service provider is characterized by the following attribute-value set

$$(CI,\ 9),\ (SC,\ 9),\ (PT,\ 9),\ (S,\ 9),\ (TCO,\ 5),\ (BI,\ 9) \rightarrow (CS,\ 8.8),$$

which form the information about the cloud service provider.

In order to decide the weight of factors of cloud service providers to assess their comprehensive strength, we can get the attributes rank and weight values of Table 3 by the ranking of attributes algorithm we proposed which are shown in Table 4. It shows that the factor S (solution) is very important than other factors when the given parameters are used for evaluating cloud service providers. The weights of the factor TCO and BI are the smallest ones. They are not the key factors. According to the result of ranking factors, we able to reduce flexibly the evaluation factors.

Table 5.3: User preferences and assessment for cloud service

Rank	Manufacturer	CS	CI	SC	PT	S	TCO	BI
1	IBM	8.9	10	9	9	9	4	10
2	Amazon	8.8	9	9	9	9	5	9
3	HP	8.7	10	8	9	9	6	9
4	Cisco	8.7	9	9	8.5	9	4.5	9
5	Saleforce	8.7	9	9	9	8.5	5	9.5
6	Dell	8.6	8.5	9	8.5	8.5	8.5	8.5
7	Huawei	8.6	9	8	8.5	9	8	9
8	Oracle	8.5	9	8	8.5	9	7	8
9	Microsoft	8.5	8	8.5	8.5	9	5	9
10	Google	8.5	8	10	8	8	8	7
11	Intel	8.4	8.5	8.5	8.5	8.5	7	8
12	EMC	8.3	9	8.5	9	8	5	8.5
13	SAP	8.2	8	8.5	8.5	8	7.5	8.5
14	H3C	8.2	8	8.5	9	8	5	8.5
15	ZTE	8.2	8	8.5	8.5	8	5	8.5
16	Alibaba	8.1	8	8.5	8.5	8	5	8.5
17	Fujitsu	8.0	8	8.5	8	8	5	8
18	Neusoft	8.0	8	8	8.5	8	5	8
19	Rackspace	7.8	8	7	8	8.5	7	7
20	Teradata	7.8	8	8	7.5	8	7	6
21	NEC	7.6	8	7.5	8	7.5	5	8
22	Tencent	7.6	7	8	8	7.5	6	7.5
23	Citrix	7.6	7	8	7.5	7.5	7	8
24	Lenovo	7.6	8	8.5	7.5	7	4.5	9
25	Joyent	7.3	9	8	8	6	6	8
26	Inspur	7.2	7.5	7	7.5	7.5	4	8
27	NetApp	7.2	7	8	7	7	7	6
28	Vmware	7.2	7	8	7	7	7	6
29	Akamai	7.2	7	8	6	7	8	8
30	Sugon	7.1	6	8	7	7	7.5	6
31	JNPR	7.1	8	7	7.5	7	4	7.5
32	Xtools	7.1	7	7.5	7	7	6	6.5
33	SNDA	7.1	7	7	8	7	4	7
34	Jingdong	7.1	7	7	7.5	7	6	7
35	Infor	6.9	7	7.5	7	6.5	6	7
36	Symantec	6.9	7	8	7.5	6	4	7.5
37	FastTrek	6.9	7	7.5	7	6.5	5	7
38	ChinaTelecom	6.9	7	7	7.5	6.5	5	7.5
39	800APP	6.8	7.5	7	7	6.5	4	7.5
40	DigitalChina	6.8	7	7.5	7.5	6	4	7.5
41	Netsuite	6.7	7.5	7	6	7	4	7.5
42	UFIDA	6.6	7	5	7	7.5	6	7
43	PowerLeader	6.6	6.5	6	6.5	7	7	7
44	Juniper	6.6	7	7	6.5	7	7	8
45	Ruijie	6.6	6	7	6.5	6.5	7	6
46	Kingdee	6.6	6.5	7	7.5	6	4	7.5
47	Vianet	6.6	7	7	6.5	6	7	7.5
48	Ucloud	6.6	7	7	7	6	4	8
49	RedHat	6.5	7	7	6	6	7	7.5
50	Unicom	6.4	6	7	7	6	4.5	7

Table 5.4: The ranking and weight of attributes

Ranking	Weight
	CI, SC, PT, S, TCO, BI
$S \succ SC \succ PT \succ CI \succ TCO = BI$	0.1, 0.25, 0.2, 0.35, 0.05, 0.05

5.5.4 An example of Application of the objective ranking of attributes approach in cloud service selection

We give an example to explain how to apply our model with personal preference. Table 5 and Table 6 are two information systems respectively based on the user preference dataset and the third-party objective dataset. To distinguish cloud service elements of subjective dataset and objective dataset, we use s_j *(j=1,2,$\cdots$,9)* and e_k *(k=1,2,$\cdots$,20)* to represent respectively the cloud service elements in Table 5 and Table 6. Attribute α_i *(i=1,2,3,4)* represents various parameters of cloud services. The value of attribute d is used to show the different decision results per cloud service. They are shown as follows:

Table 5.5: Users preference information dataset

	α_1	α_2	α_3	α_4	d
s_1	2	0	1	1	0
s_2	0	1	1	1	1
s_3	1	0	1	1	1
s_4	1	2	0	0	1
s_5	2	1	1	1	0
s_6	2	2	0	1	1
s_7	1	0	0	1	1
s_8	0	1	1	0	0
s_9	0	1	0	1	1

We can get the attributes rank, significance and weight values of Table 5 and Table 6 by Definition 2, 3 and Equation 1, or we get the result integrating the ranking of attributes algorithm and Equation 1. The results are shown in Table 7.

According to Equation 2, we can obtain the attributes ranking of cloud services with different values of weight coefficient β shown in Table 8. In a mathematical sense, the state transition of attributes ranking of cloud services selection from state i to state j with the change of the weight coefficient β is a stochastic process. From one value of

Table 5.6: Third-party objective dataset

	α_1	α_2	α_3	α_4	d
e_1	0	1	1	1	1
e_2	2	0	0	1	1
e_3	0	1	1	2	0
e_4	1	1	1	0	1
e_5	1	0	1	0	0
e_6	1	1	0	0	1
e_7	1	1	1	2	0
e_8	2	1	0	2	1
e_9	0	1	0	1	1
e_{10}	2	1	0	0	1
e_{11}	2	2	0	1	1
e_{12}	0	1	1	1	1
e_{13}	0	2	0	1	0
e_{14}	1	0	1	0	0
e_{15}	0	1	0	1	1
e_{16}	1	1	0	1	0
e_{17}	0	0	2	1	1
e_{18}	2	1	0	1	0
e_{19}	0	1	2	2	1
e_{20}	0	2	0	0	1

Table 5.7: The ranking, significance and weight of attributes

	Significance	Ranking	Weight
	$\alpha_1, \alpha_2, \alpha_3, \alpha_4$		$\alpha_1, \alpha_2, \alpha_3, \alpha_4$
Dataset in Table 5	0.444, 0, 0, 0.222	$\alpha_1 \succ \alpha_4 \succ \alpha_2 = \alpha_3$	0.67, 0, 0, 0.33
Dataset in Table 6	0.3, 0.45, 0.1, 0.6	$\alpha_4 \succ \alpha_2 \succ \alpha_1 \succ \alpha_3$	0.2069, 0.3103, 0.0689, 0.4138

β (is discrete) to other, the states of attributes ranking are known, it satisfies Markov Chains.

Table 5.8: Rankings for attributes selection

The value of weight coefficient	Ranking
Subjective dataset	
$\beta = 0$	$\alpha_1 \succ \alpha_4 \succ \alpha_2 = \alpha_3$
Objective dataset	
$\beta = 1$	$\alpha_4 \succ \alpha_2 \succ \alpha_1 \succ \alpha_3$
Comprehensive datasets	
$\beta = 0.1$	$\alpha_1 \succ \alpha_4 \succ \alpha_2 \succ \alpha_3$
$\beta = 0.3$	$\alpha_1 \succ \alpha_4 \succ \alpha_2 \succ \alpha_3$
$\beta = 0.5$	$\alpha_1 \succ \alpha_4 \succ \alpha_2 \succ \alpha_3$
$\beta = 0.7$	$\alpha_4 \succ \alpha_1 \succ \alpha_2 \succ \alpha_3$
$\beta = 0.9$	$\alpha_4 \succ \alpha_2 \succ \alpha_1 \succ \alpha_3$

5.6 Experiments result and analysis

The experiment has two goals. The first one aims for sorting the parameters of cloud services according to their significance to guide the new cloud service users to make decision. The second one aims to prove the method is effective in the application of the cloud services selection with preference information. Due to lack of the related standard test platform of users' preference and the standard test datasets, here we adopt data sets (download from the UCI [121]) as the training samples to carry out. Beside that, the original datasets are pre-processed to be easily used for calculating and program designing.

Table 9 shows the basic information of the data sets. Programming code is by Java language. It is executed sequentially on a processor Intel Core2 Duo CPUs x64. The main function of the algorithm is to give the importance order of the attributes. We can get the comprehensive weights of attributes according to the result of ranking and significance of attributes. We can get the ranking attributes by setting the different values of weight coefficient β. Thus we compare to the services matching rate successfully. The experiment regards the objective datasets as the benchmark for analysis to draw graphic. Services matching is used to describe the intention of the selection of cloud users for cloud services providers. We can get the result shown in Figure 4 for the example in section 5.

It can be seen from Figure 4 that with weight coefficient β greater, users' subjective preference play a primary role, and the service match-making rate decreases; rather, combining the subjective data and objective data, the cloud service match-making rate

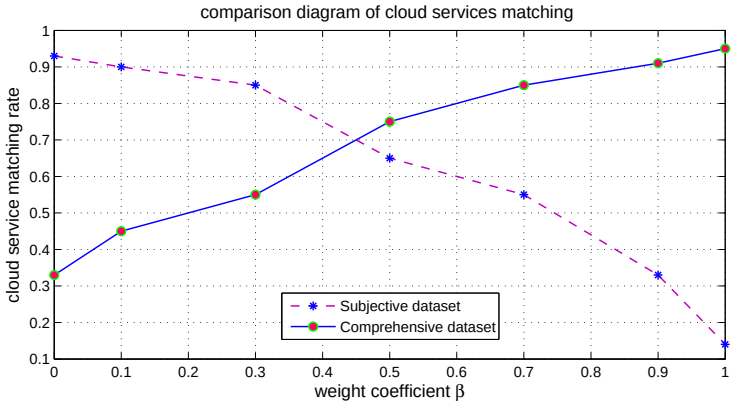


Figure 5.4: Cloud services match-making with various value of β

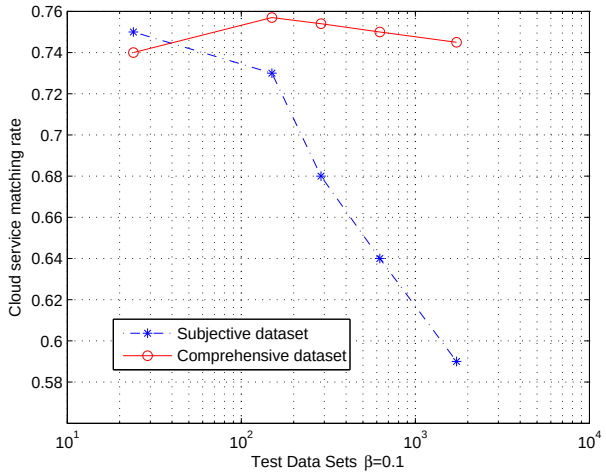


Figure 5.5: Cloud services match-making with varies data sets

increases.

Table 5.9: Basic information test data sets

Data sets	1	2	3	4	5
Number of Attributes	5	5	7	5	7
Number of Objects	24	150	287	625	1727

The users with the different subjective preference of the attribute weight use the random data to get the subjective service matching rate. As mentioned above, we use the rough set methods to get the objective weight of the attribute, integrating the objective and subjective weight to get the comprehensive matching rate of the service. Here, we set weight coefficient β is 0.1, 0.3, 0.5, 0.7 and 0.9 separately. The results are shown in Figure 5∼9.

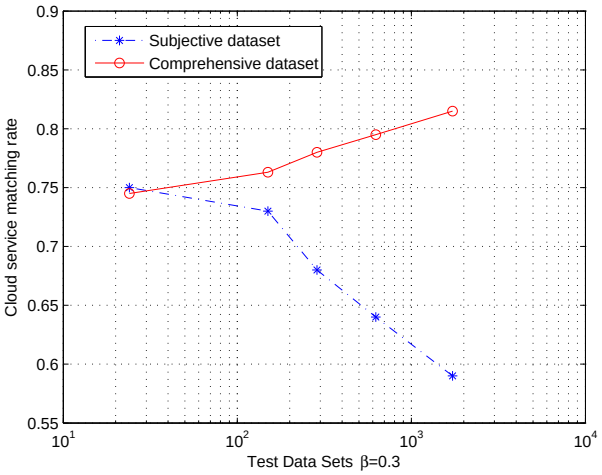


Figure 5.6: Cloud services match-making with varies data sets

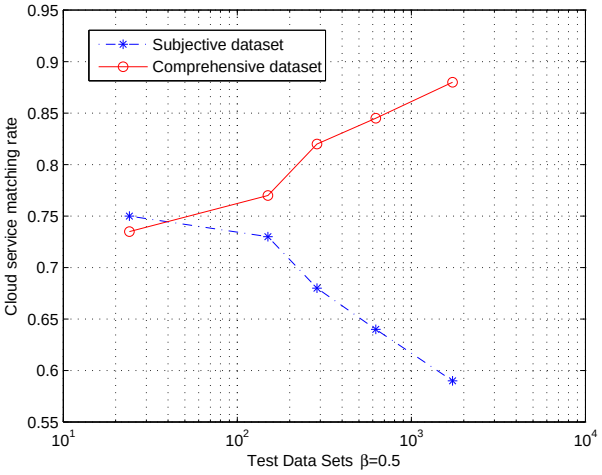


Figure 5.7: Cloud services match-making with varies data sets

We can see in Figure 5∼9, when the dateset have less service objects, the comprehensive selection or subjective selection has high service matching rate successfully. With the data increases, the comprehensive weight matching rate increases, whereas the cloud service match-making rate decreases based on the subjective preference information.

In [106], the author proposed an analytical framework to explore the significant factors affecting the adoption of SaaS for enterprise users using rough set theory. The main contribution is to mine the important factors. Although our work is similar to it in context, but our study goes to one step further, mining the significant factors in assessing cloud service providers (shown in Table 3), for example. There are six factors (*CI, SC, PT, S, TCO, BI*) in the information system of cloud service provider. It can mine four factors (*CI, SC, PT, S*) which are the important influence factors for

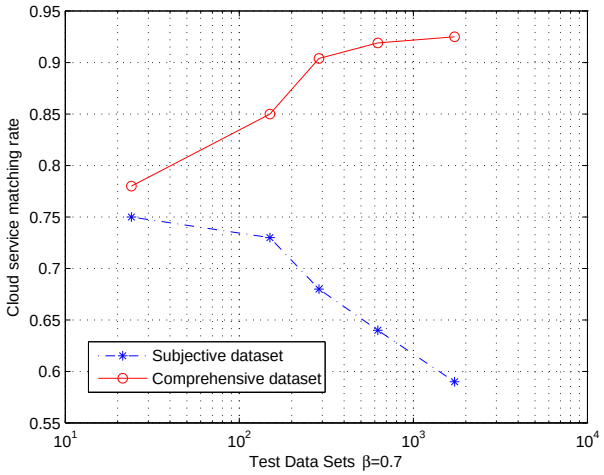


Figure 5.8: Cloud services match-making with varies data sets

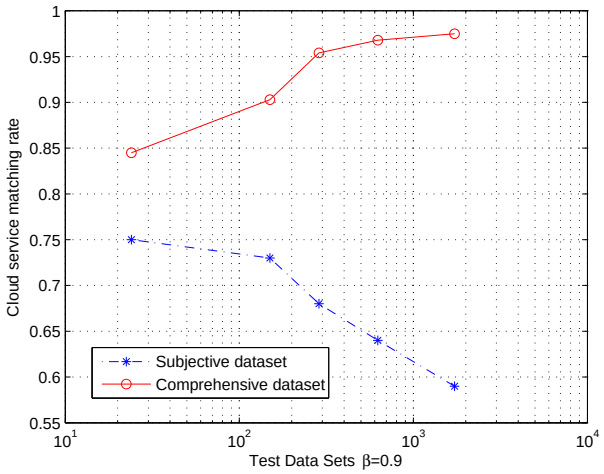


Figure 5.9: Cloud services match-making with varies data sets

evaluating the cloud service providers using the approach in [106]. Beyond that, we can't get the additional information about the result. However, in our study, we not only can know which factor is the important evaluation index of cloud service provider assessment but also rank them according to their weight, as the result shown in Table 6. Further, we can define a threshold to select evaluation factors at a stretch based on the result to design the evaluation system. In table 6, we suppose that, for some reason, we need to reduce the number of evaluation factors from 6 to 4. The method in [106] and ours both are effective. That is, the factors *TCO* and *BI* would be removed because their influence is smaller than others for evaluating cloud service providers. And if, we need to reduce the number of evaluation factors from 6 to 3, first, we remove the two factors (*TCO, BI*), after that, we don't know which factor would be removed

among the other four factors (*CI, SC, PT, S, TCO, BI*) based on the approach in [106], because there are no more information to guide us to do further. Therefor, the method proposed in [106] is failed in this case. However, in our work, beside removing the two factors (*TCO, BI*), we can judge easily to remove the factor (*CI*), because its weight is lower than the other factors', or according to the rank of factors importance shown in Table 6.

5.7 Conclusion

To provide a guide choosing the appropriate cloud services for cloud users, we present the rank-making of the parameters importance in cloud services selection and propose a ranking attributes method based on the rough set theory. It can explore the significant factors affecting the adoption of cloud services for users. At the same time, it can help the cloud service providers to specifically improve their quality of services to win more customers. We use rough set theory in the design of the algorithm to rank the parameters of cloud services. Then we can get the different weights of attributes of cloud services from subjective dataset and objective dataset. Our experimental results show that our approach is effective in services matching. Our future work will focus on optimizing the cloud services selection with more complex preferences. In the next chapter, we will summarize our works and put forward the future works.

Chapter 6

Conclusions and future works

6.1 Conclusions

With the development of cloud computing, more and more services are provided from the Internet. Cloud services provide many benefits for cloud users. The cloud service can be accessed anywhere with Internet and provides virtually unlimited resources for cloud users. At the same time, more and more cloud service providers arise. Many of them provide the same kind of services. For an enterprise or a personnel facing with so many increasing cloud services, it is difficult for them to choose the services. Especially, when many different cloud providers offer the same kind of services with different characteristics, this problem becomes even severe. The cloud users require to choose the most suitable service from so many cloud providers with lower price and shorter time. However, most of the cloud users don't know the details of the cloud services. In fact, they don't need to know many details about a service. They are only interested in some specific properties of service according to their own requirements. For example, when a small enterprise want to extend its business and the enterprise information management system cannot meet the new requirements, it will turn to the help of cloud services. Because the cost of purchasing new computer hardware and software is too expensive. And it will consume many human resources to maintain these devices. It is attractive for small enterprise to deploy the business in cloud servers in a way of pay-as-you-go. This can reduce the investment for new business. All the enterprises need to do is to choose the best cloud services from different providers with some requirements, such as the requirement of operating systems, cpu and storage. But the decision process is quite difficult for the cloud users.

In order to solve this problem, we propose a decision augmentation technology for cloud users. It can help cloud users choose the best services effectively and accurately. In

this paper, we firstly propose a decision support framework based on rough set theory. Then we use it to process the information about users' preference and properties of services. Finally, we prove that the framework can provide the most suitable choice for cloud users.

The contributions of the thesis are as follows.

Firstly, we survey the state-of-the-art methods and tools in cloud services selection area. According to the purpose of our research and the problems we solve, we use the rough set theory as our research tool. The rough set theory is a new data mining tool and has shown to be useful in many research areas.

Secondly, we propose a cloud service selection method based on rough set theory. Our method can fully use the benefits of rough set theory. We introduce in details how to use rough set theory in the research area of cloud service selection. We first propose a cloud service selection framework based on the rough set theory. The framework gives in details about how to obtain the input data, how to reduce the data information and how to generate selection rules. The final output of the framework is an auxiliary or suggested selection results. Then the cloud users can make the final decision based on their preferences and the auxiliary selection results. The final section result is reasonable since it take into considerations the objective selection result from our proposed framework and the subjective preference from cloud users.

Thirdly, we propose a parameter estimation method for cloud service. We use this method to provide reference suggestions for both cloud users and cloud providers. The parameters of cloud services are vital important for cloud providers. Since the parameters of cloud services reflect the main interests for cloud users selecting the cloud services. In order to have more advantages in market competition, cloud providers can have better understanding of the demands of cloud users with our parameter estimation method. Therefore, we propose the cloud service evaluation method. We take into consideration several common evaluation criterion. The weights of these criterion are give by experts and can be user-defined. These weights are called subjective criterion. On the other hand, we consider some other evaluation criterion whose weights are defined by the rough set theory based method. These weights are called objective criterion. Our proposed parameter estimation method is based on the subjective and objective criterion.

Finally, we design the experiments to evaluate our proposed methods. We use different sets of input data to test. It shows that our proposed method can choose the suitable cloud services for cloud users. The rate of cloud services match-making are increased.

6.2 Future works

In the future, we will extend the cloud service selection framework based on rough set theory by introducing complex criterion for data reduction. We will consider hierarchy analysis for complex estimation parameters. Analytic hierarchy process (AHP) can be introduced in our method to deal with these problems. Every criterion can derive many sub-criterion. All the criterion distributes to different layers. When executing data reduction using rough set theory, only criterion with the same layer can be used to data reduction. We can generate decision rules for each layer. Then we can reduce the generated decision rules with rough set theory. Finally, we can get the suggested decision. For extra large scale data sets, we can introduce the concept of modularity. We will first deal with the blocked data sets and then union and reduce the blocked decision rules. With the increasing of data size, the execution time will increase too. In order to improve the scalability of our proposed method, we will move forward to redesign the data reduction algorithm. At the same time, we will reduce the time complexity and improve the accuracy of the algorithm.

We will consider the optimal selection for cloud service composition. Service composition can fully utilize the current services. The optimal composition of these services in order to meet the needs of cloud users is a challenging problem now. Because more complex criterion will be introduced by the consideration of cloud service composition.

The recourse allocation in cloud computing is another challenging problem. Rough set theory, as one of the powerful tools in data mining, can be used to predict the resource usages in cloud servers. It can use the log data about the jobs and resources to pre-fetch some cloud resources for corresponding cloud services, which can improve the quality of cloud services.

Summary of Thesis in French

Résumé de la thèse en français

Sélection de services cloud en utilisant la théorie des ensembles approximatifs

Introduction (Chapitre 1)

Le Cloud computing est un domaine qui connait un véritable essor ces dernières années. Il offre de nombreux avantages pour les entreprises et les organisations en rendant les services liés à l'informatique moins onéreux et plus accessibles aux non experts. Quand ils contractent des services de cloud computing, tels que des applications logicielles, le stockage de données, ainsi que les capacités de traitement des données, les entreprises peuvent améliorer leur efficacité et leur capacité à répondre plus rapidement et de manière fiable aux besoins de leurs clients. Le Cloud computing permet en effet d'offrir aux utilisateurs des services rapides, fiables et innovants.

Les utilisateurs des services cloud n'ayant plus à investir dans l'infrastructure informatique, l'entretien des équipements, l'achat et la mise à jour du matériel ou des logiciels, ce qui leur permet de réduire les coûts et de déployer rapidement des solutions personnalisées et flexibles. Ainsi, l'utilisation du cloud permet aux entreprises de libérer des ressources et du temps pour se concentrer sur l'innovation et le développement de nouveaux produits et services. En outre, les fournisseurs de services cloud qui se sont spécialisés dans un domaine particulier peuvent apporter des services avancés qu'une entreprise ne serait pas en mesure de payer ou de développer en peu de temps. Cependant, comme toute nouvelle technologie, le cloud computing doit faire face à un certain nombre de des défis dont les plus importants sont les suivants : 1) la sécurité et le respect de la vie privée, 2) la facturation, 3) l'interopérabilité et la portabilité, 4) la fiabilité et la disponibilité, 5) les performances réseau et la bande passante. Pour répondre à ces défis, un nombre conséquent de travaux de recherche a été initié et des résultats obtenus mais les chercheurs continuent à explorer cette voie de recherche très prometteuse.

Les services cloud permettent à l'utilisateur de réduire ses coûts mais comme les fournisseurs de services cloud sont de plus en plus nombreux, les utilisateurs du cloud doivent pouvoir choisir les fournisseurs les plus appropriées. Cependant, Cette tâche s'avère très complexe pour une entreprise. L'objectif de notre travail consiste donc à aider les utilisateurs de services cloud à choisir les fournisseurs les plus appropriés mais également à permettre aux fournisseurs de services cloud d'améliorer la qualité de leurs produits et services.

Description de la problématique et des solutions envisagées

Le processus de prise de décision est difficile, que cette décision concerne l'acquisition d'une maison, l'organisation d'un voyage, ou tout simplement le choix du film à voir. Il l'est davantage pour les entreprises, qui envisagent de déplacer une partie de leurs données dans le cloud, car il en va de leur développement voire de leur survie face à la rude concurrence. L'utilisateur du cloud ne voit pas bien sûr pas toute cette complexité ni la rude concurrence entre les fournisseurs du marché. Les coûts des services cloud computing sont, dans l'ensemble, peu onéreux mais comme les données peuvent être stockées à l'étranger, il est possible que les lois du pays où les données sont hébergées permettent potentiellement à des gouvernements ou des organisations tierces d'accéder à des données relatives aux activités des utilisateurs, remettant en cause la confidentialité de données et de leur usage. Les utilisateurs du cloud doivent donc choisir un fournisseur de service assurant le niveau de sécurité correspondant le mieux à la nature et sensibilité de leurs données.

Notre travail vise à évaluer les services cloud ou leurs fournisseurs pour aider les utilisateurs dans leur prise de décision. Il est très difficile de développer une évaluation complète des fournisseurs de services cloud sans avoir défini au préalable une certaine structure ou un cadre. Ainsi, les problèmes que nous devons résoudre sont les suivants : 1) définir une manière permettant d'établir un cadre pour extraire des informations utiles afin d'aider les utilisateurs à prendre la bonne décision; 2) identifier une méthode permettant d'évaluer l'importance des paramètres utilisés pour sélectionner les services cloud. Pour résoudre ces problèmes, nous avons tout d'abord besoin de choisir les techniques d'exploration de données (data mining) appropriées. Parmi les techniques les plus courantes de data mining, on peut citer le clustering, les arbres de décision, les réseaux de neurones, etc. Dans notre étude, nous avons choisi la théorie des ensembles approximatifs comme outil de recherche sachant que ce choix sera argumenté ultérieurement. Le cadre permettant d'évaluer les fournisseurs de services cloud en utilisant la théorie des ensembles approximatifs ainsi que l'approche utilisée pour évaluer l'importance des paramètres et de les classer afin de procéder à la sélection de services cloud seront également détaillés.

Objectifs de la thèse

Cette recherche a les objectifs suivants:

a) élaborer un cadre pour la sélection de services cloud en utilisant la théorie des ensembles approximatifs basée sur une matrice de confusion (discernibility matrix) pour extraire des règles qui aident les utilisateurs du cloud dans leur prise de décision.

b) évaluer l'importance des paramètres des services cloud et les classer en utilisant la théorie des ensembles approximatifs

c) Comparer notre proposition aux travaux de la littérature.

Les techniques de sélection de services cloud (Chapitre 2)

Dans ce chapitre, nous introduisons des concepts de base tels que le cloud computing et la composition de services. Le but de ce chapitre est de présenter les techniques et algorithmes de classification existants. En fait, nous ne visons pas à comparer les avantages et inconvénients de toutes les techniques de classification mais plutôt de montrer la pertinence du choix de la théorie des ensembles approximatifs comme outil de recherche permettant de résoudre la problématique abordée dans cette thèse. À cet égard, ce chapitre sera dédié à la description, plutôt sommaire, des techniques de classification et des raisons motivant le choix de notre approche. Les défis entourant la sélection de services cloud seront également présentés ainsi que les travaux de la littérature qui s'y ont intéressés.

Le cloud computing

Le NIST (National Institute of Standards and Technology) définit le cloud computing comme suit:
Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [21].

Le cloud computing permet un modèle de consommation des services informatiques de type "pay as you go" semblable au modèle de fournisseurs de gaz, d'électricité et d'eau, selon lequel, une fois les utilisateurs de cloud y sont connectés, ils peuvent consommer autant de services qu'ils le souhaiteraient et payer pour les ressources consommées [22]. Des ressources telles que le stockage, l'accès au réseau, aux plates-formes informatiques sont provisionnés en tant que services. L'utilisation des ressources et l'efficacité opérationnelle peuvent être améliorées grâce au partage des ressources de calcul. Le prix que devra payer l'utilisateur sera inférieur en passant par un fournisseur de services cloud que s'il devait le faire lui-même (déploiement de l'application, configuration des paramètres, hébergement de l'application, etc.).

Les modèles de déploiement

Selon son type de déploiement, le cloud peut avoir des ressources privées limitées comme il peut avoir accès à de grandes quantités de ressources accessibles à distance. Les modèles de déploiement présentent un certain nombre de compromis dans la façon dont les clients peuvent contrôler leurs ressources, et l'échelle, le coût et la disponibilité des ressources. En effet, on a les catégories de déploiement suivantes : 1) cloud privé, 2) cloud communautaire 3) cloud public, 4) cloud hybride, 5) cloud privé sur-site.

Généralement, les modèles de services cloud peuvent être classifiés en trois catégories:

Infrastructure as a service (IaaS) : qui consiste en la fourniture de manière virtuelle de ressources informatiques sous la forme de matériels, d'accès au réseau et de capacités de stockage. Les utilisateurs du cloud peuvent déployer et exécuter les logiciels dont ils ont besoin. L'IaaS peut également inclure la fourniture de systèmes d'exploitation et de technologies de virtualisation pour gérer ses propres ressources d'infrastructure virtuelle, qui est généralement construite par machines virtuelles hébergées par les fournisseurs IaaS [24]. Le but de l'IaaS est d'éviter l'achat et l'installation de nouvelles ressources alors que celles-ci peuvent être louées facilement.

Platform as a service (PaaS): dans ce cas, il s'agit d'un environnement abstrait et intégré basé sur le cloud computing qui prend en charge le développement, l'exécution et la gestion des applications, dans lequel les applications sont hébergées par les fournisseurs de services et mis à la disposition des clients sur Internet. Le PaaS vise à fournir de capacités de niveau supérieur nécessaires aux applications plutôt que des machines virtuelles [24]. Avec le PaaS, les fonctionnalités du système d'exploitation peuvent être modifiées et améliorées fréquemment.

Software as a service (SaaS): ne représente pas un environnement autonome car les applications et services sont souvent utilisés en combinaison avec d'autres composants et applications du cloud. Les applications SaaS des entreprises sont associées à d'autres applications et plates-formes sur leur propre centre de données et sur d'autres plates-formes de cloud computing. Les fournisseurs de services font toutes les mises à jour et correctifs tout en gardant l'infrastructure en cours d'exécution.

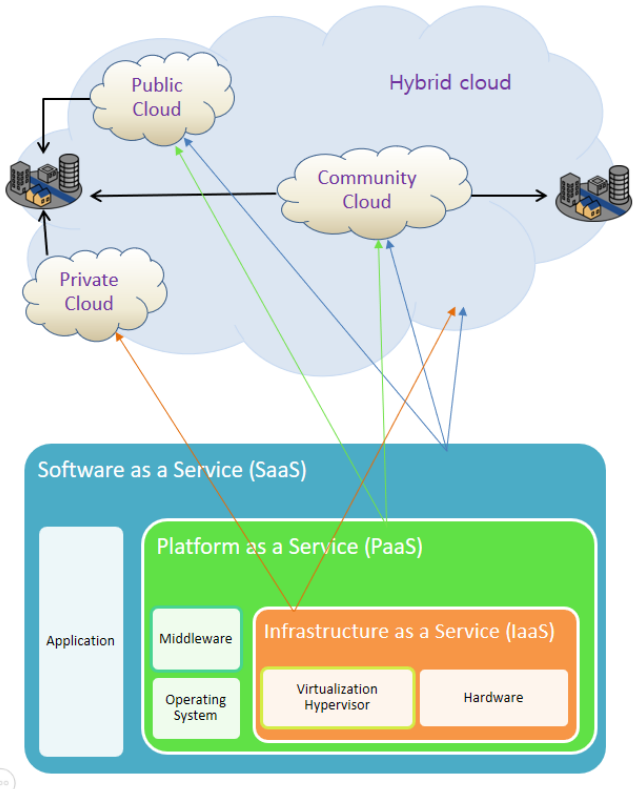


Figure 1. Déploiement de cloud computing et modèles de service

Techniques de sélection de services cloud

Beaucoup des connaissances nécessaires à la prise de décision sont cachées dans les grandes masses de données (big data) et la classification représente une forme d'analyse de données. Elle permet d'extraire un modèle qui décrit l'ensemble des données importantes ou de prédire la tendance future des données. En outre, La classification peut être utilisée pour prédire la catégorisation des données.

Parmi les méthodes de classification, on peut citer les règles d'association, la méthode des K plus proches voisins, les arbres de décision, les algorithmes bayésiens basés sur la logique floue, les algorithmes génétiques, les ensembles approximatifs, les réseaux de neurones, etc.

Un grand nombre d'algorithmes de classification sont proposés par les chercheurs qui travaillent dans les domaines d'apprentissage machine, de systèmes experts, de statistique et de neurobiologie, etc. Ces algorithmes de classification sont habituellement évalués selon des paramètres tels que la précision, la vitesse, la robuste, l'évolutivité et l'interprétation.

Il existe de nombreux algorithmes de classification et de prise de décision. Dans ce qui suit, nous présentons quelques approches telles que les arbres de décision, les réseaux Bayésiens, les règles d'association et les SVM, les réseaux de neurones et l'approche AHP.

Algorithmes de classification à base d'arbres de décision

Un arbre de décision est un outil d'aide à la décision qui utilise un graphique sous forme d'un arbre ou d'un modèle de décisions et de leurs conséquences possibles, y compris les résultats des événements, les coûts des ressources et l'utilité [25].

Les arbres de décision sont couramment utilisés en recherche opérationnelle, en particulier dans l'analyse décisionnelle, pour aider à identifier une stratégie plus susceptible d'atteindre un objectif. Les procédures d'analyse des arbres de décision peuvent répondre à des complexités de décision avec une grande incertitude, 1) il y a un nombre important de facteurs qui doivent être pris en compte lors de la prise de décision, 2) une décision de remplacement ne peut pas être prévue avec certitude, 3) considérer la possibilité de réduire l'incertitude dans la prise de décision grâce à la collecte d'informations supplémentaires [25]. Si dans la pratique, les décisions doivent être prises en ligne avec des connaissances incomplètes, un arbre de décision doit être complété par un modèle de probabilité ou par un algorithme de sélection en ligne. Une autre utilisation des arbres de décision consiste en les considérant comme un moyen descriptif pour calculer la probabilité conditionnelle.

Les algorithmes de classification à base d'arbres de décision aussi connus sous le nom d'algorithmes gloutons utilisent des heuristiques et peuvent déduire les règles de classification à partir d'un ensemble désorganisé d'instances sans règles. Les algorithmes de classification à base d'arbres de décision sont largement utilisés car ils

sont robustes même en présence de bruit et peuvent apprendre la forme normale disjonctive d'une expression logique.

Un arbre de décision est constitué de nœuds et d'arcs. Pour prendre une décision, on commence au nœud racine, et on pose des questions afin de déterminer le nœud suivant, jusqu'à ce qu'on atteigne un nœud feuille, indiquant que la décision est prise. Chaque nœud interne de l'arbre de décision représente un test sur un attribut (par exemple si une pièce va tomber sur son côté pile ou face), chaque branche représente une sortie d'essai et chaque noeud feuille représente l'étiquette de la classe ou la distribution de la classe (décision prise après le calcul de tous les attributs).

Les avantages de la classification à base d'arbres de décision de classification sont les suivants [26]:

1) Elle peut affecter des valeurs spécifiques au problème, aux décisions et aux résultats de chaque décision, ce qui réduit l'ambiguïté dans la prise de décision. Tous les scénarios possibles d'une décision sont représentés clairement, ce qui permet la visualisation claire de toutes les solutions possibles dans une vision globale.

2) Elle permet une analyse complète des conséquences de chaque décision possible, comme ce que la décision entraîne, si elle se termine dans l'incertitude ou par une conclusion définitive, ou si elle conduit à de nouvelles questions pour lesquelles le processus doit être répété. En outre, elle permet de partitionner les données dans un niveau beaucoup plus profond, pas aussi facile à réaliser avec d'autres classifieurs décisionnels tels que la régression logistique ou les SVMs.

3) Elle peut être combinée avec d'autres techniques de décision. Modèles d'arbre de décision sophistiqués sont mis en œuvre pour les applications de logiciels personnalisés, qui peuvent utiliser des données historiques pour appliquer une analyse statistique et de faire des prédictions concernant la probabilité d'événements. Par exemple, l'analyse d'arbre de décision contribue à améliorer la capacité de prise de décision des banques commerciales en attribuant le succès et la probabilité de défaillance sur les données d'application pour identifier les emprunteurs qui ne répondent pas aux critères traditionnels, minimum standards fixés.

4) Dans les classificateurs à un seul étage, un seul sous-ensemble de caractéristiques est utilisée pour distinguer parmi toutes les classes. Cette fonctionnalité sous-ensemble est généralement sélectionné par un critère globalement optimale, comme séparabilité inter-classe moyenne maximale. Dans la décision de classification d'arbres, d'autre part, on a la possibilité de choisir différents sous-ensembles de caractéristiques à différents noeuds non-terminaux de l'arbre de sorte que le sous-ensemble de la fonction choisie de manière optimale une discrimination entre les classes de ce noeud. Cette flexibilité peut effectivement apporter une amélioration des performances sur un classificateur en une seule étape.

5) Il se concentre sur la relation entre les divers événements et de ce fait, réplique le cours naturel des événements, et en tant que telle, reste solide avec peu de place pour les erreurs, à condition que les données sont correctes.

Les inconvénients de l'arbre de décision de classification:

1) La fiabilité des informations contenues dans l'arbre de décision dépend des informations d'alimentation interne et externe précis dès le début. Même un petit

changement dans les données d'entrée peut parfois provoquer des changements importants dans l'arbre. La modification des variables, à l'exception des informations de la duplication ou la modification de la mi-chemin de la séquence peut conduire à des changements majeurs et pourrait éventuellement nous exiger redessiner l'arbre.

2) Les décisions contenues dans l'arbre de décision sont fondées sur les attentes et anticipations irrationnelles peuvent conduire à des défauts et des erreurs dans l'arbre de décision. Bien que l'arbre de décision suit un cours naturel des événements en traçant relations entre les événements, il est impossible de prévoir toutes les éventualités qui découlent d'une décision, et les oublis peuvent conduire à de mauvaises décisions.

3) Les arbres de décision, tout en fournissant des illustrations faciles à voir, peuvent aussi être difficiles à manipuler. Même les données qui est parfaitement divisées en classes et qui utilisent uniquement des tests de seuil simples peuvent nécessiter d'un grand arbre de décision. Les grands arbres ne sont pas intelligibles, et se posent des difficultés de présentation.

4) Il peut y avoir des difficultés impliquées dans la conception d'un classifieur optimal d'arbre de décision. La performance d'un arbre de classification de décision dépend fortement de la façon dont l'arbre est conçu.

5) Pour les données, y compris les variables catégorielles avec un nombre différent de niveaux, le gain de l'information dans l'arbre de décision est biaisé en faveur de ces attributs avec plusieurs niveaux.

Classification bayésienne

Bayes classifieur repose sur l'application du théorème de Bayes avec des hypothèses d'indépendance entre les fonctions. Ce classifieur est nommé d'après Thomas Bayes (1702-1761) [29], qui a proposé le théorème de Bayes.

Classification bayésienne fournit des algorithmes d'apprentissage pratiques et connaissances antérieures et données observées peuvent être combinées. Classification bayésienne offre une perspective utile pour comprendre et évaluer de nombreux algorithmes d'apprentissage [30]. Il calcule les probabilités explicites pour un hypothèse et il est robuste aux bruits dans les données d'entrée.

L'idée principale de Bayes classifieur est un rôle d'une classe de prédire les valeurs de caractéristiques pour les membres de cette catégorie. Des exemples sont regroupés en classes parce qu'ils ont des valeurs communes au travers des caractéristiques. Ces classes sont souvent appelées espèces naturelles. Si un agent connaît la classe, il peut prédire les valeurs des autres caractéristiques. Si elle ne connaît pas la classe, la règle de Bayes peut être utilisée pour prédire la classe compte tenu des valeurs de caractéristiques. Dans un classifieur bayésien, l'agent d'apprentissage construit un modèle probabiliste des caractéristiques et utilise ce modèle pour prédire la classification d'un nouvel exemple.

Les avantages et les inconvénients de Bayes classifieur sont comme la suite:

- Rapide à former (seul balayage)

- Rapide pour classer
- Non sensible aux caractéristiques non pertinentes
- Poignées données réelles et virtuelles
- Gère les données de transmission en continu et discret
- En supposant l'indépendance des fonctions

Classification basée sur les règles d'association

Association minière de la règle est une tâche importante pour la découverte des relations intéressantes entre les variables dans les grandes bases de données. Il est un outil puissant pour découvrir les règles de l'exploration de données [34]. Association minière de la règle est présenté par Agrawal, Imielinski et Swami dans leur article de 1993 [35]. Il vise à étudier le comportement d'achat des clients pour trouver des régularités.

L'application prototype est l'analyse du panier du marché, qui est, d'exploiter les ensembles d'éléments qui sont fréquemment achetés ensemble dans un supermarché en analysant les achats des clients chariots (les soi-disant paniers du marché). Une fois que nous extrayons les ensembles fréquents, ils nous permettent d'extraire les règles d'association entre les ensembles d'objets, où nous faire une déclaration sur la façon dont les deux ensembles d'éléments de co-produisent sont susceptibles ou se produisent de manière conditionnelle. En plus de l'analyse du panier de consommation ci-dessus, les règles d'association sont utilisés aujourd'hui dans de nombreux domaines d'application, y compris l'extraction de Web d'utilisation, la détection d'intrusion, la production en continu, et la bioinformatique. Par exemple, dans le scénario de web log ensembles fréquents nous permettent d'extraire des règles comme: Les utilisateurs qui visitent les ensembles de pages principales, les ordinateurs portables et les promotions visiter également les pages shopping-chariot et le contrôle", indiquant peut-être que l'offre de rabais spécial se traduit par plus de ventes d'ordinateurs portables. Dans le cas de paniers sur le marché, nous pouvons trouver des règles telles que "Les clients qui achètent du lait et des céréales ont aussi une tendance à acheter des bananes, qui peuvent inciter une épicerie de co-localiser les bananes dans l'allée des céréales. En contraste avec l'exploitation minière de séquence, règle d'association en général ne considère pas l'ordre des éléments, soit dans une transaction ou à travers des transactions.

Machine à vecteurs de support

La méthode de machine à support vecteur (SVM) est une méthode de classification basée sur la marge linéaire discriminante qui est maximale, SVM sont basés sur le concept de plan de décision. Le but est de trouver l'hyperplan optimal qui maximise l'espace ou la marge entre les classes. Un plan de décision est celui qui sépare entre un ensemble d'objets ayant de différentes appartenances de classe. Un exemple

schématique est présenté dans Figure 2. Dans cet exemple, les objets appartiennent soit à la classe bleu ou la classe rouge. La ligne de séparation, dit classifieur dans la suite, définit une limite sur la côté droite de tous les objets qui sont bleu et à gauche de laquelle tous les objets sont rouges. Tous les nouveaux objets (cercles blancs) se positionnant à droite (gauche) du classifieur sont classés comme BLUE (RED).

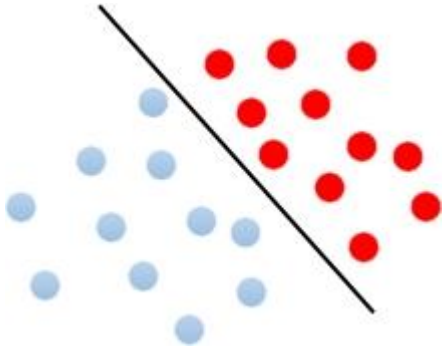


Figure 2. Un classificateur linéaire

La figure 2 est un exemple classique d'un classificateur linéaire, à savoir, un classifieur qui sépare un ensemble d'objets dans leurs groupes respectifs (bleu et rouge dans ce cas) avec une ligne. La plupart des tâches de classification, cependant, ne sont pas aussi simple que cela, et souvent des structures plus complexes correspondantes sont nécessaires afin de faire une séparation optimale, à savoir classer correctement les nouveaux objets (cas du test) sur la base des exemples qui sont disponibles (cas de l'apprentissage). Cette situation est présentée dans Figure 3. Par rapport au schéma précédent, il est clair que la séparation complète des objets bleus et les objets rouges exigerait une courbe (qui est plus complexe qu'une ligne linéaire). La tâche de classification basée sur le dessin des lignes de séparation des objets de différentes appartenances de classe est connu comme la classification en cherchant des hyperplanes. Support Vector Machines sont particulièrement adaptés à ces tâches.

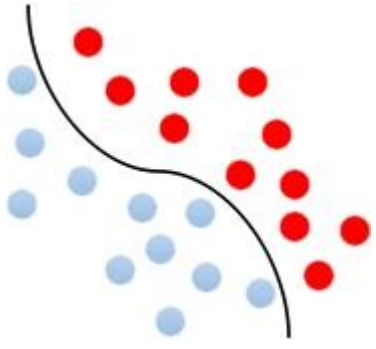


Figure 3 classificateurs hyperplanes

Support Vector Machines (SVMs) sont avant tout une méthode classique qui exécute des tâches de classification par la construction des hyperplans dans un espace multidimensionnel qui sépare les objets de différentes classes. SVM prend en charge

les tâches de régression et de classification et peut gérer des variables continues et catégorielles multiples.

Les algorithmes génétiques

Les algorithmes génétiques (GA) sont des algorithmes adaptatifs de recherche heuristique basée sur les idées évolutionnistes de la sélection naturelle et de la génétique dans le domaine de l'intelligence artificielle. Il est proposé par la Hollande en 1975 [94]. La technique de base de l'algorithme génétique est conçu pour simuler des processus dans les systèmes naturels nécessaires à l'évolution. Cet algorithme est généralement utilisé pour générer des solutions utiles à l'optimisation et la recherche des problèmes. Il exploite l'information historique pour diriger la recherche dans la région de la meilleure performance au sein de l'espace de recherche.

Les algorithmes génétiques simulent la survie du plus fort chez les personnes de beaucoup de générations consécutives pour résoudre un problème. Chaque génération est constituée d'une population de chaînes de caractères qui sont analogiques au chromosome. Chaque individu représente un point dans un espace de recherche et une solution possible. Les individus de la population sont ensuite mis à un processus d'évolution.

La procédé de fonctionnement de base de l'algorithme génétique se présente comme suit:

a) Initialisation: Réglage de la génération de l'évolution contre t = 0, fixé la génération de l'évolution maximale T, M individus générés aléatoirement comme population initiale P (0).

b) L'évaluation individuelle: le calcul de la remise en forme de chaque individu dans la population P (t). \\ Un score de la remise en forme est attribué à chaque solution représentant les capacités d'un individu à ses concurrences.

c) L'opération de sélection: le but est de choisir les individus optimales ou de nouveaux individus produits par éplucher et croiser dans la prochaine génération. L'opération de sélection est basée sur l'évaluation de l'aptitude des individus d'une population.

d) Opération de Crossover: opérateur de croisement joue un rôle important dans les algorithmes génétiques.

e) L'opération de mutation: pour changer la valeur génétique de certaines chaînes de caractères individuels dans la population. Population P (t) évolue vers la prochaine génération de la population P (t + 1) par la sélection, le croisement et l'exploitation de mutation.

f) La condition de terminaison: si t = T, sortir la solution que l'individu avec une condition physique maximale et résilier le calcul.

L'organisme de l'algorithme génétique est présenté dans Figure 4.

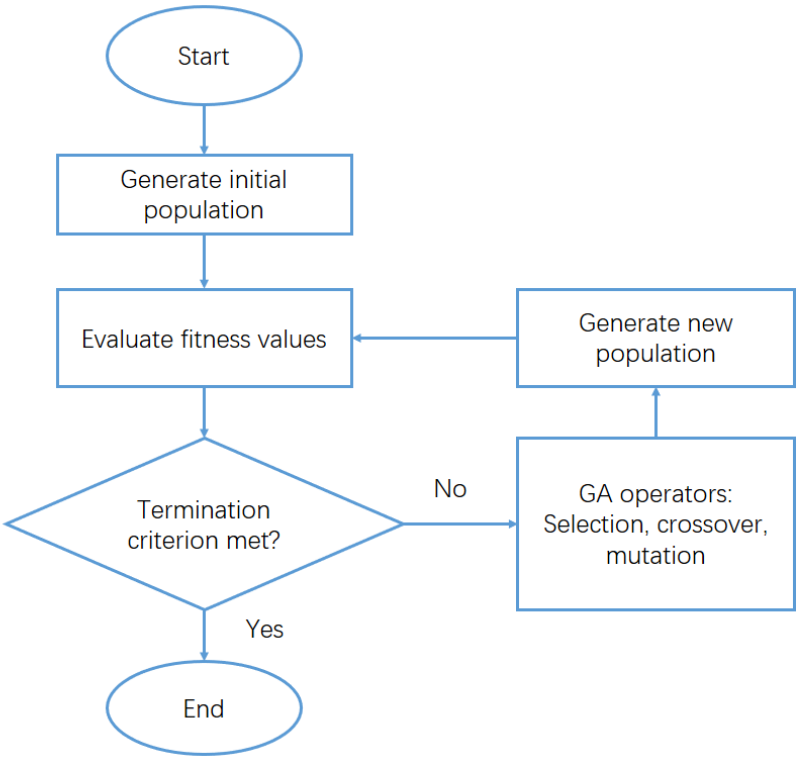


Figure 4. Algorithme génétique organigramme

Les caractéristiques des algorithmes génétiques sont ci-dessous:

1) Agir directement sur la structure de l'objet, et il n'existe pas la continuité de la dérivée de la fonction définie.

2) Parallélisme hérité implicit mondial et les meilleures capacités d'optimisation.

3) Méthode probabiliste de l'optimisation qui peut etre obtenue automatiquement et le guide optimisé de l'espace de recherche adaptative qui sert à ajuster la direction de recherche, la règle ne nécessite de la déterminaison en avance.

Il y a des limites de l'algorithme génétique:

1) L'évaluation répététive de la fonction de remise en forme pour les problèmes complexes est souvent le facteur le plus prohibitif et limité des algorithmes évolutionnaires artificiels. Trouver une solution à des problèmes complexes de grande dimension, multimodals nécessite souvent des évaluations très coûteuses de la fonction de remise en forme.

2) Les algorithmes génétiques évoluent mal avec la complexité. Autrement dit, lorsque le nombre d'éléments exposés à la mutation est grande, il y a souvent une augmentation exponentielle de la taille de l'espace de recherche. Il est donc extrêmement difficile d'utiliser la technique sur des problèmes tels que la conception d'un moteur, d'une maison ou d'un avion. Afin de rendre ces problèmes faisables à la recherche de l'évolution, ils doivent être ventilés dans la représentation la plus simple possible.

3) Dans de nombreux problèmes, l'algorithme génétique peut avoir une tendance à converger vers un optimum local ou même des points arbitraires plutôt que l'optimum

global du problème. Cela signifie qu'il ne "savent" pas comment consacrer la remise en forme à court terme pour gagner la remise en forme à plus long terme.

4) Opérer sur des ensembles de données dynamiques est difficile, car les génomes commencent à converger plus tôt vers des solutions qui ne sont plus valables pour les données ultérieures.

5) Algorithme génétique ne peut pas résoudre efficacement les problèmes dans lesquels la seule mesure de la remise en forme est une vraie / fausse mesure (comme les problèmes de décision), car il n'y a aucune moyen de converger vers la solution (pas côte à monter).

6) Pour la spécification des problèmes d'optimisation et des instances de problèmes, d'autres algorithmes d'optimisation peuvent être plus efficaces que les algorithmes génétiques en termes de vitesse de convergence.

AHP

Processus analytique de l'hiérarchie (AHP) est une technique de décision structurée pour décomposer les éléments connexes de prise des décisions à partir des objectifs, des directives, des programmes de différents niveaux afin de faire une analyse qualitative et quantitative. Il a d'abord été proposé par Thomas Saaty dans les années 1970 et est largement utilisé dans de nombreux environnements de décision. Au lieu de fournir une décision correcte, le processus analytique de l'hiérarchie essaye de trouver la meilleure décision qui correspond à la compréhension des décideurs. Pour utiliser le processus, les décideurs doivent d'abord décomposer le problème de décision en plusieurs sous-problèmes indépendants. Dans le processus de prise des décisions, les décideurs peuvent en faire une partie, en faisant leurs propres jugements. Cela signifie que les jugements subjectifs des individus peuvent avoir une grande influence sur le processus de prise des décisions.

Le processus de prise des décisions pour le processus analytique de l' hiérarchie est la suivante:

1) Modéliser le problème de décision comme une hiérarchie. Préciser le but de la décision, les alternatives et les critères.

2) Établir des priorités parmi les éléments de l'hiérarchie en faisant une série de jugements en fonction des comparaisons en paires.

3) Synthètiser ces jugements pour donner une vision globale des priorités de l' hiérarchie.

4) Vérifier la cohérence des jugements.

5) Prendre une décision finale basant sur des résultats de ce processus.

Les avantages de la méthode de hiérarchie multicritère sont comme suit.

1) En premier lieu, elle concerne une procédé d'analyse systématique. Le processus analytique de l'hiérarchie prend en compte les problèmes de décision en tant que le système. Le résultat final est influencé par tous les facteurs dans le système. Les poids de chaque couche du système modifient directement ou indirectement le résultat final.

Cette méthode est adaptée à l'évaluation des objectifs multiples, multi-critères et multi-périodes.

2) En second lieu, il est assez simple et facile à utiliser. Il transforme les multi-buts problèmes en multi-hiérarchies avec des buts simples, qui peuvent largement simplifier le calcul. Il est facile pour faire comprendre les décideurs.

3) Troisièmement, il a besoin de moins d'informations quantitatives. Il simule le chemin de la façon dont les gens prennent des décisions en laissant des informations importantes pour les cerveaux. Cela économise le calcul des frais généraux et par conséquence, résoud de nombreux problèmes pratiques qui ne peuvent pas être résolus par l'optimisation classique.

Les inconvénients de la méthode de l'hiérarchie multicritère comprennent:

1) D'abord, il ne peut pas fournir la nouvelle politique sur la prise des décisions. Le processus analytique de l'hiérarchie permet de sélectionner la meilleure politique parmi les candidats. Toutes les politiques sont connus auparavant. Le processus analytique de l'hiérarchie ne propose pas une politique nouvelle de forme différente en comparaison des candidats.

2) Deuxièmement, de nombreux facteurs qualitatifs existent donc il est difficile de croire à une simple décision. Il prend en compte de nombreux facteurs qualitatifs en simulant le processus de prise de décision des cerveaux humains.

3) Troisièmement, les statistiques se développe avec des critères.

Le processus analytique de l'hiérarchie est très utile pour les groupes qui ont des problèmes complexes. Il peut résoudre le problème des décisions bien même si les éléments importants de la décision ne sont pas précis. Le processus analytique de l'hiérarchie a été largement utilisé dans des situations de décision complexe. Il peut être appliquée dans les cas suivants: premièmement, le choix de la décision, le processus analytique de l'hiérarchie permet de sélectionner la meilleure politique à partir d'un ensemble de candidats; deuxièmement, lorsqu'il n'y a pas une seule méilleure décision, comment comparer les choix (dont la méthode de faire le(s) choix est appelée classement : triant tous les candidats en fonction de certains critères); troisièmement, la gestion de la qualité. Le processus analytique de l'hiérarchie mesure les différents aspects de la qualité.

Les défis de la sélection des services dans le cloude

La sélection de service Cloud est un sujet impliqué dans des discussions très variées. Dans les environnements de cloud computing distribués et en évolution constante, il y a de nombreux défis, tels que (i) un système automatisé recommandé par une sélection de service correspondant en permanence, (ii) le service approprié sera choisit selon les besoins des utilisateurs, pour satisfaire les besoins des utilisateurs de cloud entrants dans la composition des services en nuage, donc la collaboration entre les courtiers et les fournisseurs de services est nécessaire, (iii) le classement des multiples services ou d'optimiser la composition des services sont également des problèmes clés, (iv) la détermination de l'importance des paramètres de

services de cloud computing et de la sélection des fournisseurs de services de cloud computing.

Les approches existantes pour la sélection de cloud services

ABC (colonie d'abeilles artificiel) sont largement adopté pour trouver une solution approximativement optimale à l'état restreint. Dans la sélection des services en nuage, une stratégie en voisinage de ABC est utilisée pour améliorer la qualité de la recherche locale.

Le Bee Colony Discrete gbest guidée artificielle (DG-ABC) est un algorithme ABC amélioré, ce qui permet de simuler la recherche de la solution de composition de service optimal à travers l'exploration des abeilles pour se nourrir. Pour les données à grande échelle, il peut obtenir une solution quasi optimale avec le moins de temps.

GA (algorithme génétique) et l'algorithme génétique amélioré (IGA) sont utilisés pour l'optimisation de la composition des services en nuage. fiche Hiresome-histoire est une approche heuristique, utilisée pour traiter les problèmes de sélection de cloud services. Chaos Control algorithme optimal (CCOA) est appliqué dans la composition des services en nuage pour fournir la solution optimale.

SAW (additif pesage simple) approche est utilisée pour recommander les services de cloud computing optimales selon le calcul de leur poids.

AHP (processus de hiérarchie analytique) est de construire la structure hiérarchique pour analyser le problème. Il est appliqué dans de divers domaines de recherche. Ceci est une méthode subjective de haut niveau pour résoudre les problèmes connexes.

MADM (attribut multiple méthodologie de décision) est un ensemble de méthodes pour aider à la prise des décisions, le classement ou la sélection parmi plusieurs alternatives, dont chacun a plusieurs attributs. Il dépend d'une matrice, appelée matrice d'évaluation, matrice de décision, matrice de gain, ou une table d'évaluation.

La théorie des ensembles approximatifs est une technique d'exploration de données. Il peut explorer les informations cachées dans les grandes séries de données, par conséquent, il est un outil pour l’aide à la prise des décisions.

Connaissances liées à la théorie des ensembles approximatifs （Chapitre 3）

Avec le développement des technologies, des données et des informations d'informatique et des informations de réseau dans les divers domaines s’amplifient rapidement. Comme la participation de l'être humain et l'incertitude entre les données et l'information deviennent de plus en plus importants, les relations entre eux se compliquent. Sachant que les ressources de données et d'informations utiles et disponibles sont en abondance, nous trouvons l’importance sur la façon d’obtenir les connaissances utiles parce que les méthodes d'extraction des informations efficaces sont en pénurie, surtout dans les grandes données. Nous devrions tirer toutes les données et des informations dans la base de données de petites ou de grandes

entreprises ou des institutions. Par conséquent, la façon de traiter les grandes volumes de données sont floues, imprécises et incomplètes pour obtenir la connaissance potentiellement nécessaire, innovante et utile, il est un défi.

La théorie des ensembles approximatifs

La théorie des ensembles approximatifs, introduite par Pawlak dans le début des années 1980, est devenue un outil important de Soft Computing. La théorie des ensembles approximatifs a une capacité d'analyser qualitativement et correctement pour exprimer efficacement les connaissances incertaines et imprécises. Elle a été largement utilisée dans l'apprentissage automatique, la génération des règles, l'analyse des décisions, le contrôle intelligent dans différents domaines. Surtout, il a un grand succès dans le domaine de l'exploration de données. Les principales caractéristiques de séries brutes sont sa rigueur et robustessedes avec des définitions mathématiques strictes. Le traitement de l'information avec la théorie des ensembles bruts, sauf besoins spécifiques, ne nécessite pas de conditions préalables supplémentaires.

Système d'Information

Définition 1 $T=(U,A,V,f)$ étant un systeme informatique, ou $U = \{ X_1, X_2, \ldots ,X_n \}$ est l'ensemble fini des objets. $A=C \cup D$ est l'ensemble des attributs dont C désigne l'ensemble des attributs conditionels et D l'ensemble des attributs décisionels; $V= \cup V_\alpha$ représente l'ensemble des valeurs des attributs $\alpha \in A$; f, la fonction informatique, qui fait correspondence des valeurs aux différents attributs pour chaque objet.

Connaissances et de l'espace de la connaissance

La connaissance peut être le résumée en fonction du traitement de l'information, de l'interprétation, de la sélection et de la transformation. Il peut également être catégoriée par l'ensemble des propositions et des réglementations. En général, il est divisé en connaissance illustrative, procédurale et contrôlée. Les connaissances illustratives fournissent les concepts et les faits, par exemple, dans un système de recherche intelligent, il illustre la base de données pour des faits réels; l'utilisation des règles pour représenter les problèmes est appelée la connaissance procédurale, le plus souvent, elle est utilisée pour résoudre les problèmes posés par les connaissances illustratives dans un système de recherche intelligente; les connaissances contrôlées, y compris tous les types de traitement, des stratégies et des structures pour adapter la solution pour l'ensemble du problème. Ici, nous décrivons d'abord le modèle de la connaissance abstraite loin de la base de données avec le droit, roman et la valeur de l'application potentielle à faire comprendre les gens.

Dans la théorie des ensembles approximatifs, la connaissance est liée avec le modèle de la classification différente pour le monde réel ou subjectif. Tout objet peut être décrit par la connaissance. On peut classer les objets en fonction de la connaissance (différents attributs ou caractéristiques des objets). La connaissance est considérée comme la capacité de la classification des objets ou la connaissance lui-même, qui peut être représentée par l'ensemble de système de connaissances.

Relation d'Indiscernabilité

Définition 2 (relation d'indiscernabilité)

Étant donné un univers *U* et une grappe de relation d'équivalence S (il représente partition) en *U*, si $P \subseteq S$ et $P \neq \emptyset$, alors $\cap P$ est aussi une relation d'équivalence en *U*, elle est appelée la relation de l'indiscernabilité en *P*, notée *IND (P)* ou *P*. et ce

$$\forall x \in U, [x]_{IND(p)} = [x]_P = \bigcap_{\forall R \in P} [x]_R$$

U / IND (P) = {[x] IND (P) | ∀x ∈ U} représente les connaissances liées à la relation d'équivalence *IND (P)*, appelée P-set de base liése à l'univers *U* dans l'espace de la connaissance *K = (U , S)*. Sans confusion, *P*, *U* et *K* sont claires, nous pouvons remplacer P par *IND (P)* et *U / IND (P)* avec *U / P*. classes d'équivalence de *IND (P)* sont appelées catégories élémentaires de connaissances *P*.

Les ensembles d'approximation plus bas et l'ensemble d'approximation supérieur sont utilisés pour les concepts de base de la théorie des ensembles approximatifs. Rugueuse analyses de la théorie des jeux sont basées sur deux approximations. L'approximation inférieure et supérieure sont définies comme suit:

Le rapprochement inférieur (3,1) et le rapprochement supérieur (3,2) du sous-ensemble *X* sur la connaissance *R* sont définis respectivement par [116] [118] de la manière suivante,

$$\begin{aligned} \underline{R}(X) &= \{x|(\forall x \in U) \wedge ([x]_R \subseteq X)\} \\ &= \cup\{Y|Y \in U/R) \wedge (Y \subseteq X)\} \end{aligned} \tag{3.1}$$

$$\begin{aligned} \overline{R}(X) &= \{x|(\forall x \in U) \wedge ([x]_R \cap X \neq \emptyset)\} \\ &= \cup\{Y|Y \in U/R) \wedge (Y \cap X \neq \emptyset)\} \end{aligned} \tag{3.2}$$

Où, *[x] R* indique une classe d'équivalence de l'objet x sur les connaissances *R*. *U / R* indique les concepts élémentaires de la base de connaissances *K*.

Set $Pos_R(x) = \underline{R}(X)$ est appelé région positive;

$Bn_R(X) = \overline{R}(X) - \underline{R}(X)$ est appelée région limitée;

$Neg_R(X) = U - \overline{R}(X)$ est appelée région négative.

évidemment, $\bar{R}(X) = Pos_R(x) \cup Bn_R(X)$.

L'ensemble du rapprochement inférieur est l'ensemble de tous les objets de l'univers *U* ont certainement appartenu à l'ensemble *X* sur l'univers *U* selon les connaissances R; l'ensemble de rapprochement supérieur consiste en un rapprochement inférieur fixé et les objets de l'univers *U* ne peuvent pas être assurés dans l'ensemble *X* selon les connaissances *R*. La région limitée $Bn_R(X)$ est constituée par des éléments de l'univers *U*, qui ne peut non plus être assurée dans l'ensemble *X* selon les connaissances *R*; La région négative $Ng_R(x)$ est constituée par des éléments de l'univers *U* pas dans le jeu *X* selon *R*. de la connaissance Les approximations inférieures et supérieures du jeu *X* et la région limitée se montrent sur la figure 5.

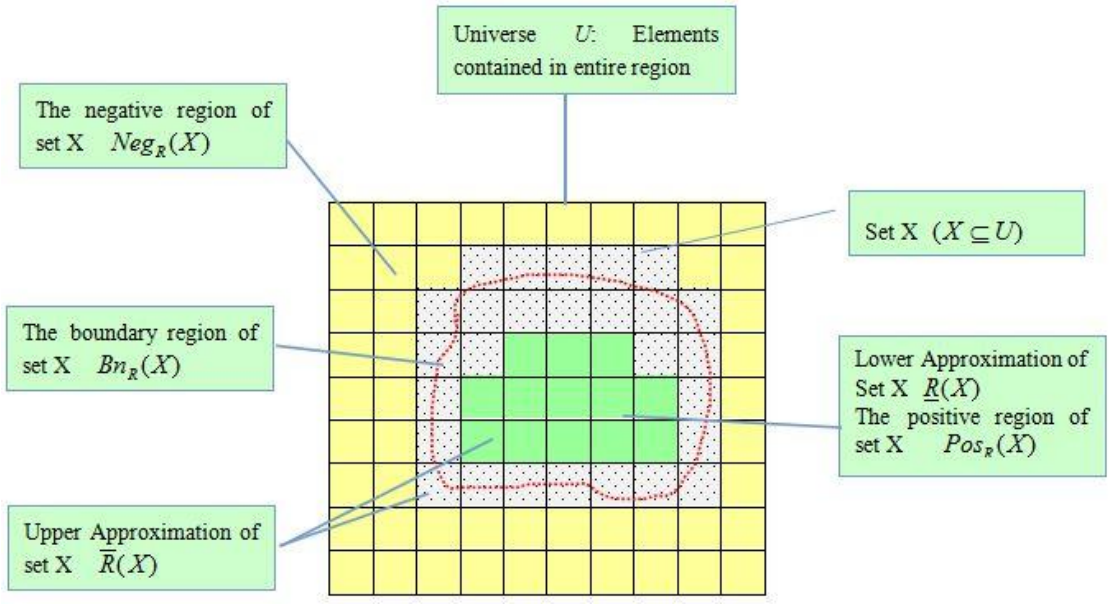


Figure 5. Les approximations inférieure et supérieure de Ensemble *X*

La réduction de la connaissance

La réduction de la connaissance est importante dans le processus intelligent. Il est l'un des contenus des bases de la théorie des ensembles approximatifs. En général, les attributs et les relations d'équivalence dans la base de connaissances ne sont pas tous aussi importants: même pour certaines connaissances nécessaires, la redondance existe. Des moyens de réduction de connaissances qui maintiennent la capacité de la classification des attributs sont définis pour supprimer la connaissance inutile.

Définition 3 Soit une base de connaissances $K = (U, S)$ et un pôle de relation d'équivalence $P \subseteq S$, $\forall R \in P$, si

$$IND(P) = IND(P - \{R\})$$

Alors la connaissance *R* est la redondance à *P*, le reste *R* est nécessaire de *P*. Si chaque $R \in P$, *R* est nécessaire de *P,* alors *P* est indépendant, sinon *P* dépend de *P*.

Théorème 1 Si la connaissance P est indépendante, $\forall G \subseteq P$, alors G est indépendant aussi.

Définition 4 **(**réduction de la Connaissance**)**

Donner un base de connaissances $K= (U, S)$ et un pôle de relation d'équivalence $P \subseteq S$, pour tout $G \subseteq P$, si G satisfait aux deux conditions:

(1) G est indépendant;

(2) $IND (G) = IND (P)$.

alors G est une réduction de la connaissance P, il est donné par $G \in RED (P)$,

dans lequel, $RED (P)$ représente la réduction du jeu de P.

Définition 5 **(**connaissances de base**)**

Compte tenu d'une base de connaissances $K = (U, S)$ et une relation d'équivalence pôle $P \subseteq S$, pour tout $R \in P$, si satisfait R

$IND (P - \{R\}) 6 = IND (P)$

Alors R est nécessaire de P, l'ensemble a consisté en connaissances nécessaires pour P appelé noyau de P, est donné par $CORE (P)$.

Théorème 2 $CORE = \cap RED (P)$

Théorème 2 démontre que le noyau de la connaissance est l'intersection de toutes les réductions de connaissance, ce moyen de base de connaissances est conclute à chaque réduction de la connaissance et peut être calculée directement. En plus de cela, le noyau de la connaissance ne peut être réduit, sinon, il serait faible la capacité de la classification des connaissances.

Extraction de Règles

L'extraction des règles de système d'expression de la connaissance est l'une des principales tâches dans le domaine de l'exploration de données et la découverte de connaissances. Normalement, quatre types de règles peuvent être extraites à partir de données, la caractéristique, l'association, le discriminante, et les règles de classification [5]. Les règles induites du rapprochement inférieur du concept décrivent certainement le concept, d'où ces règles sont appelées certaines. D'autre part, les règles induites à partir de l'approximation supérieure de la notion décrivent le concept éventuellement, de sorte que ces règles sont appelées possible.

Application de la théorie des ensembles approximatifs dans la sélection des services Cloud (Chapitre 4)

Avec la prolifération rapide des fournisseurs de services de cloud computing, il est difficile pour les utilisateurs de cloud de savoir quels sont les bons choix pour leurs besoins. De même, les fournisseurs de services de cloud computing ont besoin pour améliorer leurs services pour attirer de plus en plus d'utilisateurs de cloud computing.

Ici, nous allons donner une approche pour protéger les intérêts des utilisateurs de nuages et les fournisseurs de services cloud.

Pour les fournisseurs de services de cloud computing, le défi majeur est d'exploiter les avantages du cloud computing pour gérer la qualité des engagements de services aux clients tout au long du cycle de vie d'un service. Les utilisateurs cherchent à obtenir le service de cloud au prix le plus bas . Il y a beaucoup de services de cloud avec les fonctions identiques ou similaires, mais avec des qualités différentes. En outre, le service de cloud est un environnement dynamique et ouvert. Les événements se produisent souvent comme l'augmentation ou la diminution dynamiquement des services de cloud, la défaillance du service ou le changement. Ainsi, les utilisateurs doivent non seulement d'évaluer la qualité du service, mais aussi l'équilibre entre la qualité de service et leurs inconvénients. Ces services sont utilisés pour acheter des services de cloud computing afin de faire le bon choix. Cependant, une variété de facteurs peuvent influencer le choix du service en nuage de l'utilisateur. De nombreux utilisateurs sont préoccupés par des questions telles que la fiabilité, la disponibilité, la rapidité, tandis que d'autres soucis pour le prix et l'intégrité. Par conséquent, ils sont souvent empêtés par quel est le genre de services de cloud computing le plus approprié pour eux. Il en faut des outil d'aide à la décision.

Le choix des outils dans l'étude de la sélection des services cloud

L'effet de l'algorithme de classification ou de l'approche prise des décisions en général est lié aux caractéristiques des données parce que cet ensemble de données a des valeurs nulles, le bruit, la distribution clairsemée, ou parce que leurs valeurs d'attribut sont différents, certains sont continus, certains discrets, ou mélangés. Les classificateurs classiques sont utilisés avec succès dans de nombreux domaines divers. L'arbre de décision de classification a été appliquée dans les laboratoires de diagnostic médical, analyste financier, d'évaluer le risque de crédit de prêt demandeur; SVM (support de machine à vecteurs) a été appliqué dans la reconnaissance des formes, l'analyse génétique, la classification de texte, la reconnaissance vocale, l'analyse de régression; Le neuronal algorithme de classification du réseau est largement utilisé dans la reconnaissance optique de caractères, la biologie moléculaire, la reconnaissance du visage, parce que ce ne sont pas sensibles aux bruits des données. Comme chaque outil de l'algorithme de classification ou la prise des décisions a ses avantages et ses inconvénients, et a cause de la diversité des données et de la complexité des problèmes pratiques, il est difficile de dire ce qui est meilleur que l'autre. Par exemple, le réseau neuronal est un algorithme d'apprentissage basé sur le principe de minimisation du risque empirique, il existe une certaine faiblesse inhérente. Cependant, l'algorithme compense les SVM. Donc, en pratique, le choix de la bonne classification est essentielle pour des problèmes spécifiques.

À commencer par la recherche de la satisfaction de la demande des utilisateurs de services de cloud, nous prenons en considération des divers facteurs, puis nous

choisissons la théorie des ensembles approximatifs comme l'outil de recherche. La méthode de jeux approximatifs est une technique d'exploration de données bien connu ayant des avantages intéressants. En fait, la théorie des ensembles approximatifs ne dépend pas d'une connaissance de l'expérience, mais il repose sur des données. Il traite des informations imprécises, incertaines ou incomplètes sans la connaissance d'introniser les règles a priori qui sont utilisées pour prendre les décisions pertinentes. Il est non seulement une maniere d'aider les fournisseurs de développer leurs offres de services, mais aussi une aide pour les utilisateurs à choisir le service de cloud computing avec rentabilite adaptée à leurs besoins. Ici, la première question que nous sommes intéressés est les préoccupations qui permettent aux utilisateurs de choisir le service de cloud en utilisant la théorie des ensembles approximatifs. Cette dernière fournit de bonnes propriétés pour la découverte et la simplification des facteurs impliqués dans le choix des utilisateurs.

Nous proposons une solution de départ pour les indicateurs de système de service de cloud basés sur la théorie des ensembles approximatifs. nous déterminons d'abord les facteurs cruciaux de choisir toutes sortes de services de cloud computing pour les utilisateurs. Nous définissons les éléments de services de cloud computing comme un ensemble d'objets, les facteurs tels que les attributs de ces objets, les valeurs des attributs des objets sont les données pertinentes recueillies. Sur cette base, nous établissons le système d'information. Ensuite, nous utilisons la théorie des ensembles rugueuse pour réduire les attributs et d'exploiter les règles qui aideront les utilisateurs à prendre des décisions sur la sélection d'un service de cloud approprié.

Un cadre de la théorie des ensembles approximatifs dans les services cloud

Quand il y a de nombreux services dans les nuages, les utilisateurs espèrent rapidement pour sélectionner les services à partir des ensembles de candidats correspondant. Dans cette partie, nous adoptons la théorie des ensembles approximatifs afin de construire un modèle de sélection de services en nuage pour aider les utilisateurs à prendre la décision efficace. L'idée principale consiste à calculer des approximations inférieures et supérieures sur la base des caractéristiques spécifiques d'attributs, puis fournir des règles de sélection des services.

Basé sur le flux de travail décrit dans la figure 6, on construit les ensembles de candidats de services cloud correspondants et leurs ensembles d'attributs (les métriques d'évaluation subjectifs et objectifs) pour produire le système d'information.

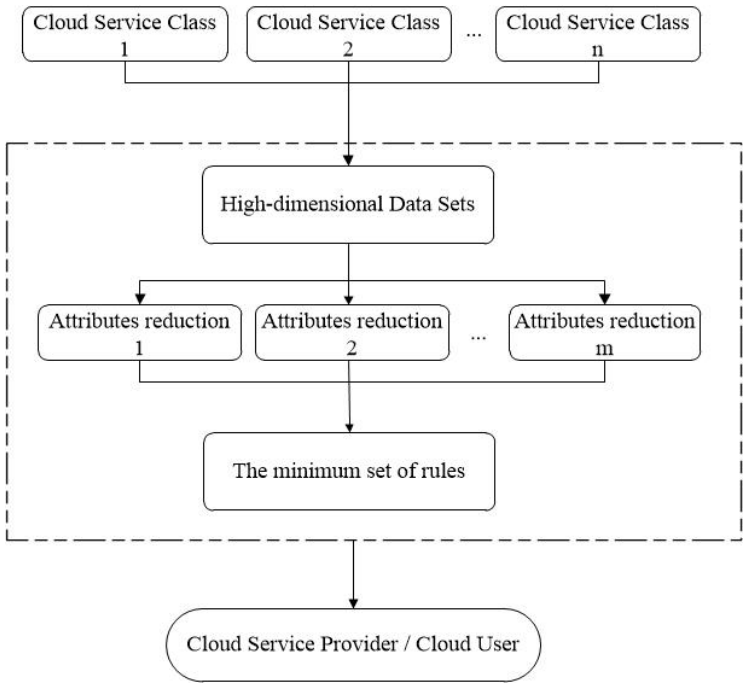


Figure 6. Sélection de services Cloud basée sur la théorie des ensembles approximatifs

Certains des tiers confidentiels et les centres de contrôle des services de cloud computing analysent les performances des services en nuage à partir des données collectées à partir des évaluations des utilisateurs de nuages. Tant que les experts combinent les caractéristiques des services de cloud computing, de nombreux paramètres peuvent être mesurés quantitativement (par exemple, la disponibilité, l'élasticité, le temps de réponse du service, et le coût par tâche). Nous pouvons évaluer et segmenter les niveaux des mesures, telles que la mémoire de lecture / écriture, le débit, la vitesse du processeur et ainsi de suite. Comme la sécurité des données de l'entreprise et la vie privée sont essentielles, elles pourraient aussi être des critères d'évaluation. Les valeurs d'attributs peuvent être extraites à partir des ensembles de date d'âme.

Les quantités massives de données brutes font habituellement les processus de décisions très compliqués. Comme les méthodes des sets approximatifs ne traitent que les attributs discrets, une série de pré-traitement tel que la discrétisation des certains attributs continus est nécessaire.

Classification et prise des décisions

Dans cette section, nous présentons les modalités d'application de la théorie des ensembles approximatifs dans la sélection de services en nuage à travers un exemple simple avec des définitions pertinentes

Voici les définitions pertinentes sur le processus de réduction des attributs et les règles d'induction:

Définition 1 Le $DT = (U, C \cup D, V, f)$ est un système d'information de décision 4-tuple, où $U = \{X_1, X_2, ..., X_n\}$ est un ensemble fini des objets et $| U | = n$. Nous

définissons la matrice de discernabilité du système d'information de décision qui suit, où *i, j = 1,2, ···, n*.

$$M_{n\times n}(DT) = (c_{ij})_{n\times n} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{bmatrix}$$

$$c_{ij} = \begin{cases} \{\alpha | (\alpha \in C) \wedge (f_\alpha(x_i) \neq f_\alpha(x_j))\}, \\ \qquad\qquad f_D(x_i) \neq f_D(x_j); \\ \emptyset, \\ f_D(x_i) \neq f_D(x_j) \wedge f_C(x_i) \neq f_C(x_j); \\ -, \qquad\qquad f_D(x_i) = f_D(x_j). \end{cases}$$

c_{ij} est l'élément dans la matrice de discernabilité.

La fonction d'information $f_\alpha(x_i)$ désigne une valeur pour l'α condition d'attribut du x_i objet. Fonction d'information $f_D(x_i)$ désigne une valeur de la décision attribut D du x_i objet.

Définition 2 [116] [118] Soit 4-tuple $DT = (U, C \cup D, V, f)$ un système d'information de décision, où $U = \{X_1, X_2, ..., X_n\}$ est un ensemble fini des objets et $|U| = n$. $\forall \alpha \in A$, $\forall X_i, X_j \in U$, nous commandons la variable de discernabilité par rapport à l'attribut α comme suit:

$$\alpha(X_i, X_j) = \begin{cases} \{\alpha | (\alpha \in C) \wedge (f_\alpha(x_i) \neq f_\alpha(x_j))\}, \\ \qquad\qquad f_D(x_i) \neq f_D(x_j); \\ \emptyset, \\ f_D(x_i) \neq f_D(x_j) \wedge f_C(x_i) \neq f_C(x_j); \\ -, \qquad\qquad f_D(x_i) = f_D(x_j). \end{cases}$$

lle est égale à l'élément C_{ij} dans la matrice de discernabilité. Donc nous avons

$$\Sigma\alpha(x_i, x_j) = \begin{cases} \alpha_{l_1} \vee \alpha_{l_2} \vee \cdots \vee \alpha_{l_k}, \\ \{\alpha(x_i, x_j) = \alpha_{l_1}, \alpha_{l_2}, \cdots \alpha_{l_k}\} \\ \qquad\qquad (1 \leq k \leq card(c); \\ -, \qquad \alpha(x_i, x_j) = \emptyset \vee -. \end{cases}$$

La fonction de discernabilité est alors définie comme suit:

$$\Delta = \prod_{\forall (x_i, x_j) \in U \times U} \sum \alpha(x_i, x_j) \overset{\text{def}}{=} \bigwedge_{\forall (x_i, x_j) \in U \times U} \sum \alpha(x_i, x_j), \quad i, j = 1, 2, \cdots, n.$$

La matrice de discernabilité et la fonction de discernabilité sont utilisées pour réduire la connaissance redondante.

Définition 3 [116] [118] Soit 4-tuple *DT = (U, C ∪ D, V, f)* un système d'information décisionnel. Soit *C, D ⊆ A*. Evidemment, si *C'⊆ C* est un *D-réduction* de *C*, alors *C'* est un sous-ensemble minimal de *C*. Nous dirons que attribut *α ∈C*, si *POSc (D) = Pos (C - {α})(D)*, puis le sous-ensemble *C' = (C - {α}) ⊆ C* est un *D-réduction* de *C*, dénoté *REDD (C)*. *CORED(C)=∩REDD(C')* sere appelée *D-noyau* de *C*.

La procédure de notre approche

Étape 1: obtenir la matrice discernabilité
Étape 2: restreindre les solutions par des attributs de réduction
Étape 3: obtenir le noyau des attributs
Étape 4: obtenir les règles

L'algorithme de réduction de la matrice de discernabilité

Algorithm 1 Attribute reduction algorithm of discernibility matrix DM

Input:

The information system of cloud services;

Output:

The attributes Reduction of the cloud services system: Red;

1: Input the information table of cloud services;
2: set $Red=\phi$, count(a_i)=0, for i=1, n;
3: compute the discernibility matrix and weight frequent of attributes count(a_i); \\ every new item C of DM, count(a_i):=count$(a_i)+n/\mid c \mid$, $a_i \in \mid c \mid$.
4: merge all the same items and order the discernibility matrix according to the length of item and frequent;
5: for each m of DM ;
6: if $(m \bigcap Red==\phi)$;
7: choose the attribute a of m, maxi=count(a);
8: $Red=Red \bigcup \{a\}$
9: end if;
10: end for;
11: return Red.

Nous testons l'algorithme avec Java. Il est exécuté sur un processeur Inter Core 2 Duo x64. nous testons tout d'abord un exemple. Le résultat montre que notre méthode est valable. Deuxièmement, nous adoptons ensembles des données (téléchargées de l'UCI [27]) pour exécuter l'algorithme, il est également valable.

Évaluation de l’importance des paramètres dans la sélection de services cloud en utilisant des ensembles approximatifs (Chapitre 5)

Depuis plusieurs années, le cloud computing a influencé le paysage informatique et devient un facteur économique important [1] en raison de sa mode de fonctionnement qui est le pay-as-you-go pour fournir un service. Depuis le cloud computing est une barrière pour l'entrée minime et la mise à l'échelle économique, il y a beaucoup de clients potentiels de passer leur entreprise à ce sujet. Dans ce contexte, de nombreux fournisseurs petits et grands de services cloud émergent chaque jour. Cependant, tous ne sont pas les propriétaires d'une infrastructure cloud au première niveau. Cela signifie que pour les fournisseurs de services de cloud computing plus petits, ils ne sont pas en partenariat avec un grand fournisseur qui possède l'infrastructure. Normalement, ce n’est pas un gros problème, même si elles sont toutes reliées à un fournisseur d'infrastructure plus grand, quand il descend, tous «agents intermédiaires» descendent avec elle. Comme les fournisseurs de services de cloud computing ont leur modèle de service spécifique, par conséquent, il est difficile pour les utilisateurs de comparer les services de cloud computing proposés par les différents fournisseurs. Par conséquent, les utilisateurs de nuages se trouvent dans un défi de choisir un fournisseur approprié en tenant compte de leurs besoins spécifiques.

Certains utilisateurs de nuages ne prennent en considération que leurs paramètres de préférences subjectives des critères d'évaluation, tout en ignorant l'importance des paramètres d'évaluation objectives obtenues à partir d'autres clients qui avaient les mêmes exigences de service quand ils choisissent les services de cloud computing. La plupart des utilisateurs de nuages ne pouvaient pas trouver un service de cloud approprié correspondant à leurs besoins individuels quand ils utilisent un service de cloud donné pour le premier. En fait, comme ils ne sont pas sûrs que la performance et la qualité du service sélectionné sont bonnes, ils choisissent sur la base de leur jugement subjectif pour les paramètres de décision adaptés. En outre, lorsque les utilisateurs de cloud essaient de donner une évaluation globale pour un service de cloud, il est également pas objective que les paramètres tels que les poids des services de cloud computing sont générés par les expériences ou les experts dont le processus marque généralement de la subjectivité. Cela influence le choix d'un service cloud adapté aux utilisateurs de cloud.

Pour toutes les questions mentionnées ci-dessus, nous pouvons obtenir la note de l'importance des attributs et de les classer par la théorie des ensembles approximatifs, ce qui nous déterminons le poids objectif des indices d'évaluation des services de cloud computing. Notre proposition peut non seulement guider les utilisateurs de nuages, face à un grand nombre de choix de services de cloud computing, concernant les indices d'évaluation(ils devraient se concentrer en davantage), mais aide également les fournisseurs de cloud computing pour améliorer la performance et la qualité des

services de cloud computing avec l'intention d'attirer plus d'utilisateurs de nuages à faire eux-mêmes qui ont une prédominance de la concurrence pour l'avenir de l'industrie des IT.

Paramètres d'évaluation des services Cloud

Le cœur de métier est varié de différents fournisseurs de services cloud. Par exemple, l'activité d'Amazon est plus intéressée par les plates-formes et logiciels (PaaS et SaaS), qui sont des services de cloud publique. Toutefois, IBM a un plus large éventail d'entreprises, dont son matériel et ses plates-formes sont plus avancés; IaaS, PaaS, SaaS et d'autres aspects de l'entreprise sont en jeu, elle est favorisée dans la construction de clouds privés et hybrides. Par conséquent, il est difficile pour l'utilisateur de définir quel service cloud fournisseurs sont les meilleurs sur la base d'un certain point. Il y a quelques paramètres de configuration pour tous les types de services de cloud computing pour évaluer leur performance. Par exemple, le système le nombre de CPU, la taille de la mémoire, l'espace de stockage, de fonctionnement et ainsi de suite, ces paramètres déterminent les performances des services de Cloud Hosting. Lorsque les utilisateurs choisissent un service cloud de type, il existe de nombreux fournisseurs de services de cloud alternatives. Lorsque les utilisateurs font leurs choix, ils ont besoin certains paramètres pour évaluer la capacité globale de fournisseurs de services de cloud computing, tels que la capacité d'innovation, la capacité de service, les technologies de produits, les solutions, l'influence de la marque. Les paramètres d'évaluation habituels de fournisseurs de services de services de cloud computing et de cloud computing sont comme suit.

1) La disponibilité de service Cloud
2) Service Cloud évolutivité
3) Service Cloud élasticité
4) La sécurité de service Cloud
5) La capacité d'innovation
6) Le cout total de la propriètè
7) La capacité de service
8) Solution
9) Marque influence

La méthode de sélection de services cloud avec des informations de préférence

Les utilisateurs du cloud donnent généralement le poids subjectif de différents paramètres du service de cloud, basant sur la préférence personnelle quand ils choisissent le service de nuage, résultant également des choix non pratiques. Par conséquent, dans cette section, nous introduisons une approche de classer l'importance des indices de services de cloud computing et de fournir le poids objectif sur les différents paramètres en fonction de la théorie des ensembles approximatifs.

Approche de classement objectif des attributs basée sur la théorie des ensembles approximatifs

Définition 1 Pour un système d'information *T=(U,A,V,f), A=C ∪D*. l'expression $PosC(D)$, nommée la région positive de la partition *U / D* par rapport a les attributs de conditions *C*, est un set de tous éléments de *U*, qui peut etre seulement classifiés en bloques de la partition *U / D* a partir de *C*. *U / D* indique les concepts élémentaires du système d'information *T* sur le set des atttributs décisionels *D*. Pour $\alpha \in C$, on a

a) Si $Pos_{C-\{\alpha\}}(D) = Pos_C(D)$, alors α est un attribut innécessaire de *C*

b) Si $Pos_{C-\{\alpha\}}(D) \neq Pos_C(D)$, alors α est un attribut nécessaire de *C*.

Définition 2 Dans un système d'information *T=(U,A,V,f), A=C ∪D*, l'importance d'un attribut du système d'information de décision peut etre testée par la capacité de classification sur T pendant le processus d'effacer un attribut conditionel du set *C*; l'importance d'un attribut est définit comme la suite par [22] :

$$Sig(\alpha) = \frac{|\,card(Pos_C(D))\,| - |\,card(Pos_{C-\{\alpha\}}(D))}{|\,U\,|} \qquad (1)$$

Card représente le cardinalité des attributs. Sig_α représente la dépendance de l'attribut décisionel *D* sur l'attribut conditionel α, qui reflète la capacité de classification sur l'attribut α. Lorsque Sig_α est plus grand, la dépendance entre l'attribut conditionel α et l'attribut décisionel *D* est plus fort, et le plus discriminative l'attribut α est.

La rigueuse analyse de la théorie des ensembles est basée sur l'espace supérieur et les approximations inférieures. Le rapprochement inférieur de l'ensemble peut se décrire par la connaissance précise dans un système d'information, qui est appelé région positive et est défini par définition 1. Si le rapprochement inférieur ne sera pas changé quand un attribut est supprimé, l'attribut est inutile et peut être réduit. Sinon, l'attribut est appelé attribut de base, ce qui est nécessaire. En d'autres termes, la définition 1 peut distinguer les principaux attributs et les attributs inutiles tout en ignorant l'effet des attributs relativement nécessaires. Pour tous les attributs relativement nécessaires, on peut les classer dans un système d'information en fonction des valeurs des attributs différents de sa signification. L'importance d'un attribut défini par définition 2 peut refléter la diversité de l'espace d'approximation inférieure lorsque l'attribut est supprimé.

Comme le service de cloud est caractérisé par de différents paramètres, tels que la disponibilité ou l'évolutivité, l'élasticité et ainsi de suite, il est difficile de définir des critères de sélection valables pour différents besoins des clients. Pour ce problème,

nous donnons une méthode de sélection de services en nuage en utilisant la théorie des ensembles approximatifs, qui est représentée dans ce qui suit:

Nous obtenons les informations subjectives de préférence des utilisateurs à travers l'interaction parmi eux. Si certains utilisateurs fournissent des informations incomplètes, nous pouvons prendre des données en mode complète ou par une traduction des informations incomplètes en remplissant un. La méthode pour obtenir des informations de préférences de l'utilisateur est représentée sur la figure 7.

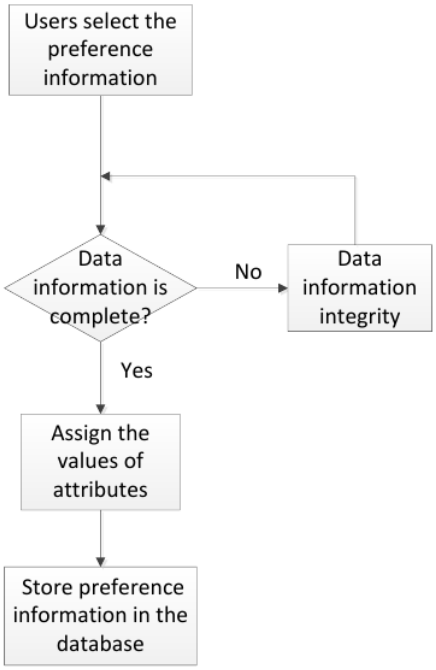


Figure 7. Obtenir l'information de préférence

Pour obtenir les paramètres d'importance de services de cloud computing, le classement des attributs algorithme est décrit dans la figure 8:

Algorithm 1 The ranking attributes of cloud services

Input:
The information system of cloud services;

Output:
The attributes ranking of the cloud services;

1: Input the information table of cloud services;
2: Set $C = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$;
3: Compute all partition *U/D* with respect to condition attributes *C*;
4: Set i=1;
5: if $i \leq$ *number of the attributes*;
then
Compute all partition *U/D* with respect to condition attributes $c = \{C - \alpha_i\}$;
i++;
6: Compute all the significance of the condition attributes with respect to decision attribute *D*

$$Sig(\alpha) = \frac{|card(Pos_C(D))| - |card(Pos_{C-\{\alpha\}}(D))}{|U|}$$

7: Rank the attributes of cloud services.

Figure 8. L'algorithme de classement des attributs de services cloud

Application du classement objectif des attributs dans la sélection des services cloud

Choisir les services de cloud computing est un problème d'attributs de prise des décisions multiple, et la clé est de déterminer le poids de paramètres. Il existe plusieurs façons de déterminer le poids d'indicateurs, en générale, qui se répartissent en deux catégories: les méthodes d'affectation subjectives et objectives. La méthode d'affectation subjective attribue les poids sur la base des informations subjectives de la prise des décisions. Il est arbitraire avec une mauvaise précision et la fiabilité de la prise des décisions. Dans la procédé d'attribution objective, chaque paramètre est évalué avec les données réelles. Dans le nuage du système de sélection de service, l'importance des attributs est différente. Le poids objectif d'attributs peut être défini comme dans (2):

$$W_{\alpha} = \frac{Sig_{\alpha}(\alpha)}{\sum_{c \in C} Sig_{c}(c)} \tag{2}$$

Le poids global en ce qui concerne les paramètres peut être défini comme dans (3):

$$I(w) = \beta W_{o}(w) + (1-\beta) W_{so}(w), 0 \le \beta \le 1 \tag{3}$$

Où, β qui est appelé le coefficient de pondération reflètant les préférences de l'utilisateur pour les poids subjectives et objectives quand ils prennent des décisions dans le choix des services de cloud computing. $W_o(w)$ et $W_{so}(w)$ représente respectivement le poids des paramètres de services de cloud computing avec l'ensemble des données objectives et subjectives. Plus petite la valeur de β indique que les utilisateurs apprécient plus leurs attributs subjectives. Inversement, plus la valeur des utilisateurs bêta souligne l'importance des paramètres objectives. Spécialement, si $\beta = 0$, le jugement de l'importance des paramètres de services de cloud computing dépendent totalement de leur prise de conscience subjective; si $\beta = 1$, les utilisateurs se fient entièrement sur le poids objectif.

Une application est illustrée pour déterminer les pondérations globales de paramètres de services cloud basés sur la théorie des ensembles approximatifs. L'obtention du poids global de chaque paramètre comprend deux parties. La première partie acquiert le poids des paramètres basés sur les données subjectives qui vienent des préférences de l'utilisateur en nuage. La deuxième partie acquiert le poids objective fondée sur les données sans information subjective du décideur. Le modèle du classement objectif des attributs dans le cloud système de sélection du service d'application est illustré à la figure 9.

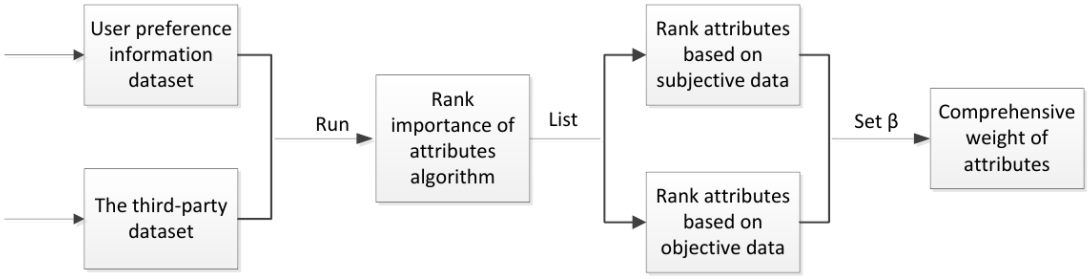


Figure 9. Modèle d'application du classement objectif des attributs

Application de l'approche de classement des attributs dans la sélection de services en nuage

Il y a des indices correspondants destinés à évaluer un système ou un service. Lorsque les fournisseurs de services de cloud computing lancent un produit de service aux consommateurs, ils doivent fournir une qualité de services et ils espèrent obtenir le feed-back des consommateurs le plus tôt possible pour améliorer leurs produits, dans le même temps, les indices d'évaluation des services soient conçus en conséquence. Pour les utilisateurs de services de cloud computing, quand ils choisissent un service de cloud, ils vont considérer certains facteurs pour obtenir le service approprié, tels que la disponibilité de services en nuage, l'élasticité du service de cloud , la marque de service, etc. Comme nous le savons, dans le marché économique, le contrôle des coûts et la poursuite de l'efficacité sont les principaux objectifs de chaque direction de l'entreprise. La raison pour laquelle les utilisateurs de cloud choisissent de transférer leurs activités vers le cloud centre de calcul est parce que cela est une bonne façon d'économiser la capital et d'améliorer l'efficacité de comparer leur modèle de développement traditionnel. Cependant, dans la pratique, les utilisateurs de cloud computing devraient équilibrer le poids des facteurs utilisés pour évaluer les services de cloud computing.

Ici, nous utilisons un example pour démontrer comment la théorie des ensembles approximatifs fonctionne sur le classement des facteurs de fournisseurs de services de cloud. La résistance globale du fournisseur de services de cloud est importante pour les utilisateurs de cloud de choisir un service approprié dans le nuage. Les données réelles dans le tableau 1 et la liste des prestataires de services de cloud computing en fonction de leur capacité est collectées en 2014. Les fournisseurs de services de cloud sont opérateurs en Chine. Les données sont publiées dans la revue de la Chine Internet de la semaine [26]. Dans le tableau 1, les facteurs tels que *CI* (capacité d'innovation), *SC* (capacité de service), *PT* (technologies de produits), S (solution), *TCO* (coût total de possession) et *BI* (influence de la marque) sont les facteurs source d'évaluation de services cloud. Le facteur *CS* (partition complète) est le résultat de l'évaluation des fournisseurs de services de cloud computing.

Tableau 1. Les scores des fournisseurs de services de cloud computing.

Rank	Manufacture	CS	CI	SC	PT	S	TCO	BI
1	IBM	8.9	10	9	9	9	4	10
2	Amazon	8.8	9	9	9	9	5	9
3	HP	8.7	10	8	9	9	6	9
4	Cisco	8.7	9	9	8.5	9	4.5	9
5	Saleforce	8.7	9	9	9	8.5	5	9.5
6	Dell	8.6	8.5	98	8.5	8.5	8.5	8.5
7	Huawei	8.6	9	8	8.5	9	8	9
8	Oracle	8.5	9	8.5	8.5	9	7	8
9	Microsoft	8.5	8	8.5	8.5	9	5	9
10	Google	8.5	8	10	8	9	8	7
11	Intel	8.4	8.5	8.5	8.5	9	7	8
12	EMC	8.3	9	8.5	9	9	5	8.5
13	SAP	8.2	8	8.5	8.5	8.5	7.5	8.5
14	H3C	8.2	8	8.5	9	8.5	5	8.5
15	ZTE	8.2	8	8.5	8.5	8	5	8.5
16	Alibaba	8.1	8	8.5	8.5	8	5	8
17	Fujistu	8.0	8	8.5	8	8	5	8
18	Neusoft	8.0	8	8	8.5	8	5	8
19	Packspace	7.8	8	7	8	8.5	7	7
20	Teradata	7.8	8	8	7.5	8	7	6
21	NEC	7.6	8	7.5	8	7.5	5	8
22	Tencent	7.6	7	8	8	7.5	6	7.5
23	Citrix	7.6	7	8	7.5	7.5	7	8
24	Lenovo	7.6	8	8.5	7.5	7	4.5	9
25	Joyent	7.3	9	8	8	6	6	8
26	Inspur	7.2	7.5	7	7.5	7.5	4	8
27	NetApp	7.2	7	8	7	7	7	6
28	Vmware	7.2	7	8	7	7	7	6
29	Akamai	7.2	7	8	6	7	8	8
30	Sugon	7.1	6	8	7	7	7.5	6
31	JNPR	7.1	8	7	7.5	7	4	7.5
32	Xtools	7.1	7	7.5	7	7	6	6.5
33	SNDA	7.1	7	7	8	7	4	7
34	Jingdong	7.1	7	7	7.5	7	6	7
35	Infor	6.9	7	7.5	7	6.5	6	7
36	Symantec	6.9	7	8	7.5	6	4	7.5
37	FastTrek	6.9	7	7.5	7	6.5	5	7
38	ChinaTelecom	6.9	7	7	7.5	6.5	5	7.5
39	800APP	6.8	7.5	7	7	6.5	4	7.5
40	DigitalChina	6.8	7	7.5	7.5	6	4	7.5
41	Netsuite	6.7	7.5	7	6	7	4	7.5
42	UFIDA	6.6	7	5	7	7.5	6	7
43	PowerLeader	6.6	6.5	6	6.5	7	7	7

Rank	Manufacture	CS	CI	SC	PT	S	TCO	BI
44	Juniper	6.6	7	7	6.5	7	7	6
45	Ruijie	6.6	6	7	6.5	6.5	7	6
46	Kingdee	6.6	6.5	7	7.5	6	4	7.5
47	Vianet	6.6	7	7	6.5	6	7	7.5
48	Ucloud	6.6	7	7	7	6	4	8
49	PedHat	6.5	7	7	6	6	7	7.5
50	Unicom	6.4	6	7	7	6	4.5	7

Dans la théorie des ensembles approximatifs, chaque fournisseur de services en nuage est représenté comme un objet de recherche, et les facteurs comme ses attributs. Parmi eux, le facteur *CS* est attribut de décision, tandis que d'autres sont les attributs de condition. Simplement, les colonnes du tableau 1 sont des attributs et les lignes sont des objets, tandis que les entrées de la table sont des valeurs d'attribut. Ainsi, chaque ligne du tableau peut être considérée comme une information sur le fournisseur de services en nuage spécifique. Notre objectif de recherche est de classer le poids des facteurs pour évaluer l'avantage global de fournisseurs de services de cloud computing.

Nous abstraite au hasard un fournisseur de services en nuage à partir du tableau 1 pour expliquer le but de nos études, par exemple, Amazon. Nous pouvons voir dans le tableau 1 que un fournisseur de services de cloud est caractérisé par l'ensemble des (attribut-valeur)s suivantes *(CI, 9), (SC, 9), (PT, 9), (S, 9), (TCO, 5), (BI, 9) → (CS, 8.8)*, qui forment les informations sur le fournisseur de services en nuage.

Afin de décider de l'importance des facteurs de fournisseurs de services de cloud computing pour évaluer leur résistance globale, nous pouvons obtenir les attributs de classement et les valeurs des poids du tableau 1 par le classement des attributs en utilisant l'algorithme que nous avons proposé, qui sont présentés dans le tableau 2. Il montre que le facteur S est assez important que les autres facteurs lorsque les paramètres donnés sont utilisés pour évaluer les fournisseurs de services de cloud computing. Les poids du facteur *TCO* et *BI* sont les plus petits. Ils ne sont pas les facteurs clés. Selon le résultat des facteurs de classement, nous faisons mesure à réduire de manière flexible les facteurs d'évaluation.

Tableau 2. Le classement et le poids des attributs.

Ranking	Weight					
	CI	*SC*	*PT*	*S*	*TCO*	*BI*
S>SC>PT>CI>TCO = BI	0.1	0.25	0.2	0.35	0.05	0.05

Résultats et analyses

L'expérience a deux objectifs. Le premier vise à trier les paramètres de services de cloud computing en fonction de leur importance pour guider les nouveaux utilisateurs à prendre une décision. Le seconde vise à prouver la méthode est efficace dans l'application de la sélection des services cloud avec des informations de préférence.

En raison de l'absence de la plate-forme du test standard liée à la préférence des utilisateurs et les jeux de données, ici nous adoptons les ensembles de données (téléchargement de l'UCI [27]) que les échantillons de formation pour mener à bien. En outre, les ensembles de données d'origine sont pré-traités pour être facilement utilisés pour le calcul et le programme de conception.

Le tableau 3 montre les informations de base des ensembles de données. Les codes de programmation est en Java. Il est exécuté de manière séquentielle sur un processeur Intel Core 2 Duo x64. La fonction principale de l'algorithme est de donner l'ordre d'importance des attributs. Nous pouvons obtenir les poids complets d'attributs en fonction du résultat de classement et l'importance des attributs. Nous pouvons obtenir les attributs de classement en définissant les différentes valeurs du coefficient de pondération β. Ainsi nous comparons les taux de services des cas de succès. L'expérience ce qui concerne les ensembles de données objectives est se référence pour l'analyse graphique. L'adaptation des services est utilisée pour décrire l'intention de la sélection des utilisateurs de cloud pour les fournisseurs des services cloud. Nous pouvons obtenir le résultat montré sur la figure 10.

Tableau 3. Informations de base des ensembles de donnéés de test.

Datasets	1	2	3	4	5
Number of Attributes	5	5	7	5	7
Number of Objects	24	150	287	625	1727

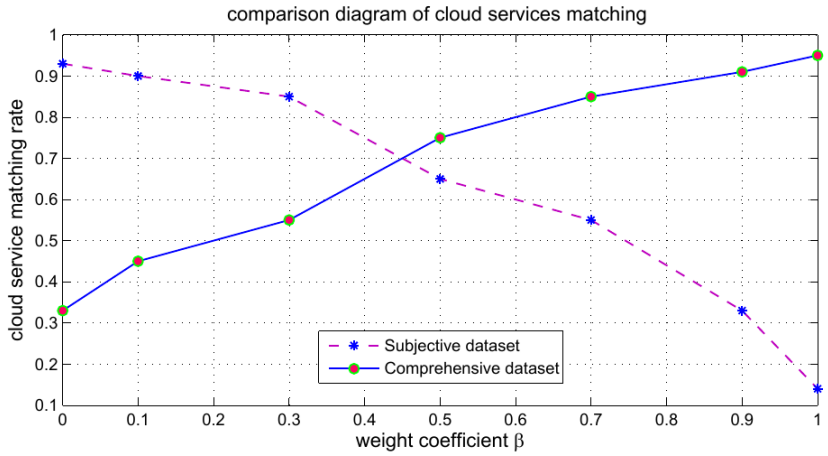


Figure 10. Couverture des services de jumelage avec de variées valeurs de β

On peut voir sur la figure 10 que, avec l'augmentation du coefficient de pondération β, la préférence subjective des utilisateurs devient plus importante, et les match-making services baissent leur taux; de plus, la combinaison des données subjectives et des données objectives font les services de cloud computing augmenter avec le taux match-making.

Les utilisateurs avec les différentes préférences subjectives du poids de l'attribut utilisent les données aléatoires pour obtenir le taux de correspondance de service subjective. Comme mentionné ci-dessus, nous utilisons les méthodes rugueuses pour

obtenir le poids objectif de l'attribut, en intégrant le poids objective et subjective pour obtenir le taux de correspondance globale du service. Ici, nous avons mis coefficient β poids 0,1, 0,3, 0,5, 0,7 et 0,9 séparément. Les résultats sont présentés sur la figure 11.

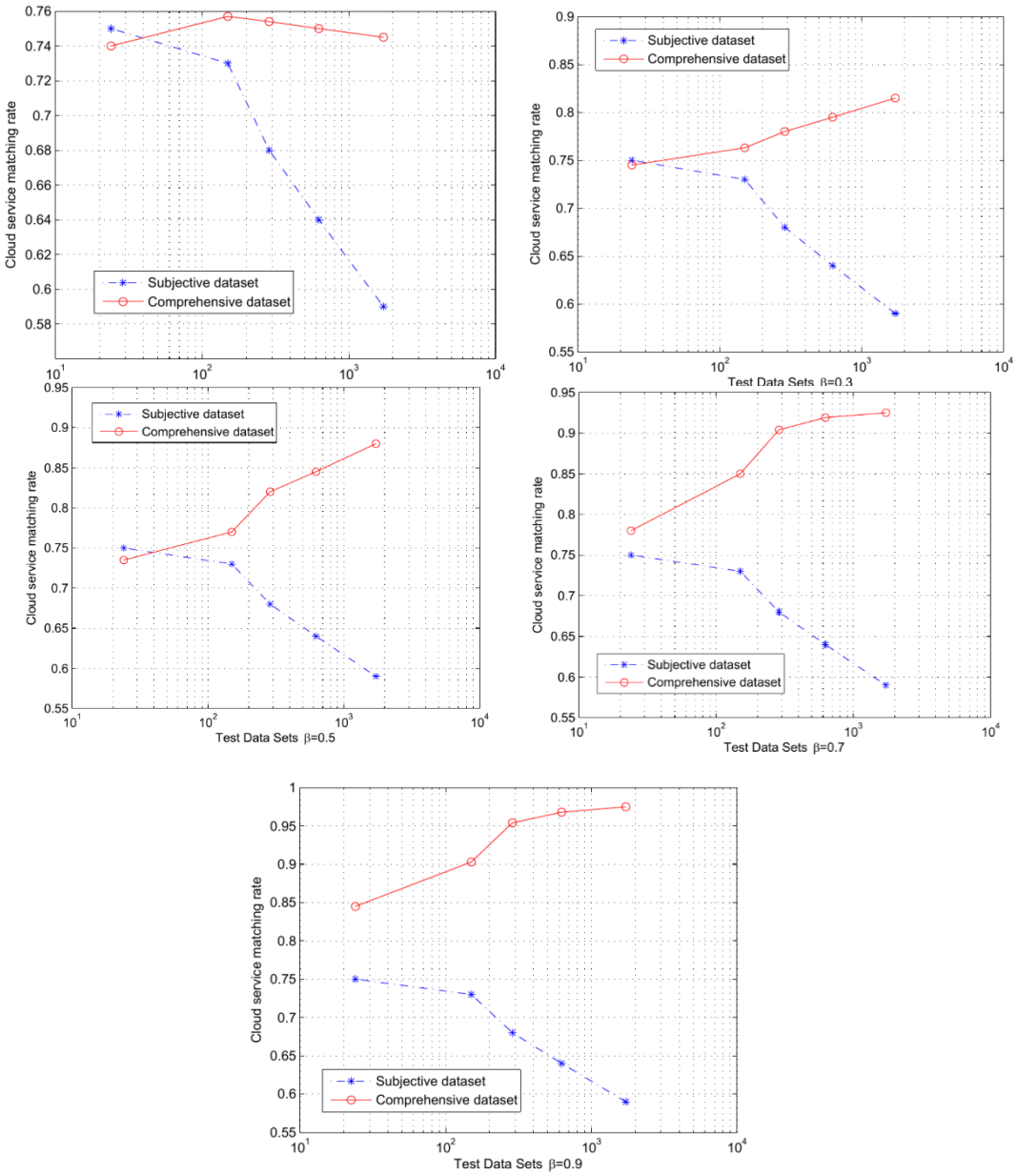


Figure 11. Service Couverture jumelage avec divers ensembles de données.

Nous pouvons voir sur la figure 11, lorsque les ensembles de données ont moins de objets de service, la sélection complète ou la sélection subjective a réussi à élever le taux d'appariement des services. Lorsque la quantité des données augmente, l'augmentation du poids global conduit à la hausse du taux de matching, alors que le taux de match-making service cloud diminue , qui ne base que sur l'information de préférence subjective.

Dans [12], l'auteur propose un cadre d'analyse pour explorer les facteurs importants qui influencent l'adoption du SaaS pour les utilisateurs de l'entreprise à l'aide de la théorie des ensembles approximatifs. La contribution principale est d'exploiter les

facteurs importants. Malgré que notre travail soit similaire dans son contexte, notre étude va un peu plus loin, l'exploitation avec des poids spécifiques des facteurs importants dans l'évaluation des fournisseurs de services de cloud computing (indiquées dans le tableau 1); par exemple, il y a six facteurs *(CI, SC, PT, S, TCO, BI)* dans le système d'information du fournisseur de services de cloud. Il peut en extraire quatre facteurs *(CI, SC, PT, S)* qui sont les facteurs d'influence les plus important pour évaluer les fournisseurs de services de cloud en utilisant l'approche dans [12]. Au-delà de cela, nous ne pouvons pas obtenir les informations supplémentaires sur le résultat. Cependant, dans notre étude, nous avons non seulement pouvons savoir quel facteur est l'indice d'évaluation important de l'évaluation de fournisseur de services de cloud computing, mais aussi les classer selon leur poids, comme le résultat montré dans le tableau 2. En outre, on peut définir un seuil pour sélectionner les facteurs d'évaluation les plus affilés en fonction du résultat de la conception du système d'évaluation. Dans le tableau 2, on suppose que, pour une raison quelconque, nous avons besoin de réduire le nombre de facteurs d'évaluation de 6 à 4. La méthode de [12] et la nôtre sont toutes efficaces. Autrement dit, les facteurs *TCO* et *BI* seraient retirés parce que leur influence est plus petit que d'autres pour évaluer les fournisseurs de services de cloud computing. Et si, il faut réduire le nombre de facteurs d'évaluation 6-3, d'abord, on enlève les deux facteurs (*TCO, BI*), après cela, nous ne savons pas quel facteur serait retiré parmi les quatre autres facteurs (*CI , SC, PT, S, TCO, BI*) à partir de l'approche dans [12], parce qu'il n'y a pas plus d'informations pour nous guider à le faire. Par conséquent, la méthode proposée dans [12] est omis dans ce cas. Cependant, dans notre travail, à part l'élimination des deux facteurs (*TCO, BI*), nous pouvons en décider facilement de supprimer le facteur (*CI*), parce que son poids est inférieur à celui des autres facteurs », ou selon le rang des facteurs d'importance indiqué dans tableau 2.

Conclusion (Chapitre 6)

Pour le but de fournir un guide sur le choix des services de cloud computing appropriées pour les utilisateurs de cloud computing, nous présentons le rang de décision de l'importance des paramètres de sélection de services cloud et proposons une méthode d'attributs classement basée sur la théorie des ensembles approximatifs. Le méthode peut explorer les facteurs importants qui influencent l'adoption de services de cloud computing pour les utilisateurs. En même temps, elle peut aider les fournisseurs de services de cloud computing à améliorer spécifiquement la qualité des services les plus personnalisées possibles. Nous utilisons la théorie des ensembles approximatifs dans la conception de l'algorithme pour classer les paramètres de services de cloud computing. Ensuite, nous pouvons obtenir les différents poids des attributs de services de cloud computing des données subjectives et des ensembles de données objectives. Nos résultats expérimentaux montrent que notre approche est efficace dans les services correspondants. Notre travail futur se concentrera sur l'optimisation de la sélection de services cloud avec des préférences plus complexes.

Les contributions de la thèse

Tout d'abord, nous intégrons les méthodes et les outils à la pointe de la technologie en nuage à la sélection des services. Selon le but de notre recherche et les problèmes que nous visons à résolver, nous utilisons la théorie des ensembles approximatifs comme outil de recherche. La théorie des ensembles approximatifs est un nouvel outil d'exploration de données et a été prouvée utile dans de nombreux domaines de recherche.

Deuxièmement, nous proposons une méthode de sélection des services de cloud computing basée sur la théorie des ensembles approximatifs. Notre méthode peut utiliser au maximum les avantages de la théorie des ensembles approximatifs. Nous présentons en détail comment utiliser la théorie des ensembles approximatifs dans la zone de recherche de la sélection de services en nuage. Nous proposons d'abord un cadre de sélection de services en nuage basé sur la théorie des ensembles approximatifs. Le cadre donne les détails sur la manière d'obtenir les données d'entrée, la façon de réduire les informations de données et la façon de générer des règles de sélection. Le résultat final de ce cadre est un des résultats des sélections auxiliaires ou suggérées. Ensuite, les utilisateurs de nuages peuvent prendre la décision finale en fonction de leurs préférences et les résultats des sélections auxiliaires. Le résultat de la dernière section est raisonnable car elle prend en considération le résultat de la sélection objectif de notre cadre proposé et la préférence subjective des utilisateurs dans le nuages.

Troisièmement, nous proposons une méthode d'estimation des paramètres pour le service de cloud. Nous utilisons cette méthode pour fournir des conseils de référence pour les utilisateurs en nuage et des serveurs cloud. Les paramètres de services de cloud computing sont vitalement importants pour les fournisseurs de cloud. Les paramètres de services de cloud computing reflétent les principaux centres d'intérêt pour les utilisateurs de cloud computing en sélectionnant les services de cloud computing. Afin d'avoir plus d'avantages dans la concurrence du marché, les fournisseurs de cloud peuvent avoir une meilleure compréhension des besoins des utilisateurs de nuages avec notre méthode d'estimation des paramètres. De plus, nous proposons la méthode d'évaluation de services en nuage. Nous prenons en considération plusieurs critères d'évaluation communes. Les poids de ces critères sont donnés par des experts et peuvent être définis par l'utilisateur. Ces poids sont appelés critères subjectifs. D'autre part, nous considérons d'autres critères d'évaluation dont les poids sont définis par la méthode basée sur la théorie des ensembles approximatifs. Ces poids sont appelés critères objectifs. Notre méthode proposée pour estimer des paramètres est basée sur les critères subjectifs et objectifs.

Quatrièmement, nous concevons des expériments pour évaluer nos méthodes proposées. Nous utilisons de différents ensembles des données d'entrée pour tester. Il montre que notre méthode proposée peut choisir les services de cloud computing appropriés pour les utilisateurs de cloud computing. Le taux de services de cloud computing match-making sont amélioré.

PUBLICATIONS

The following is a list of publications that have been published and accepted as parts of this thesis.

[1] LIU, Yongwen, ESSEGHIR, Moez, et BOULAHIA, Leila Merghem. Cloud service selection based on rough set theory. In : Network of the Future (NOF), 2014 International Conference and Workshop on the. IEEE, 2014. p. 1-6.

[2] LIU, Yongwen, ESSEGHIR, Moez, et BOULAHIA, Leila Merghem. Evaluation of parameters importance in cloud service selection using rough set theory. Applied Mathematics. Vol.7 No.6 2016.

References

[1] SUBASHINI, Subashini et KAVITHA, V. A survey on security issues in service delivery models of cloud computing. Journal of network and computer applications, 2011, vol. 34, no 1, p. 1-11.

[2] REN, Kui, WANG, Cong, et WANG, Qian. Security challenges for the public cloud. IEEE Internet Computing, 2012, no 1, p. 69-73.

[3] CHEN, Deyan et ZHAO, Hong. Data security and privacy protection issues in cloud computing. In : Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on. IEEE, 2012. p. 647-651.

[4] CARLIN, Sean et CURRAN, Kevin. Cloud computing security. 2011.

[5] RONG, Chunming, NGUYEN, Son T., et JAATUN, Martin Gilje. Beyond lightning: A survey on security challenges in cloud computing. Computers and Electrical Engineering, 2013, vol. 39, no 1, p. 47-54.

[6] SO, Kuyoro. Cloud computing security issues and challenges. International Journal of Computer Networks, 2011, vol. 3, no 5.

[7] JAMIL, Danish et ZAKI, Hassan. Security issues in cloud computing and countermeasures. International Journal of Engineering Science and Technology (IJEST), 2011, vol. 3, no 4, p. 2672-2676.

[8] FERNANDES, Diogo AB, SOARES, Liliana FB, GOMES, Joao V., et al. Security issues in cloud environments: a survey. International Journal of Information Security, 2014, vol. 13, no 2, p. 113-170.

[9] BHADAURIA, Rohit, CHAKI, Rituparna, CHAKI, Nabendu, et al. Security Issues In Cloud Computing. Acta Technica Corviniensis-Bulletin of Engineering, 2014, vol. 7, no 4, p. 159.

[10] WHAIDUZZAMAN, Md et GANI, Abdullah. Measuring security for cloud service provider: A Third Party approach. In : Electrical Information and Communication Technology (EICT), 2013 International Conference on. IEEE, 2014. p. 1-6.

[11] ZISSIS, Dimitrios et LEKKAS, Dimitrios. Addressing cloud computing security issues. Future Generation computer systems, 2012, vol. 28, no 3, p. 583-592.

[12] MOWBRAY, Miranda et PEARSON, Siani. A client-based privacy manager for cloud computing. In : Proceedings of the fourth international ICST conference on COMmunication system softWAre and middlewaRE. ACM, 2009. p. 5.

[13] YU, Yong, NIU, Lei, YANG, Guomin, et al. On the security of auditing mechanisms for secure cloud storage. Future Generation Computer Systems, 2014, vol. 30, p. 127-132.

[14] WEI, Lifei, ZHU, Haojin, CAO, Zhenfu, et al. Security and privacy for storage and computation in cloud computing. Information Sciences, 2014, vol. 258, p. 371-386.[Chapter Introduction 13]

[15] WANG, Cong, WANG, Qian, REN, Kui, et al. Privacy-preserving public auditing for data storage security in cloud computing. In : INFOCOM, 2010 Proceedings IEEE. Ieee, 2010. p. 1-9.

[16] SABAHI, Farzad. Cloud computing security threats and responses. In : Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on. IEEE, 2011. p. 245-249.

[17] FERRER, Ana Juan, HERNNDEZ, Francisco, TORDSSON, Johan, et al. OPTIMIS: A holistic approach to cloud service provisioning. Future Generation Computer Systems, 2012, vol. 28, no 1, p. 66-77.

[18] NODEHI, Tahereh, GHIMIRE, Sudeep, et JARDIM-GONCALVES, Ricardo. Toward a unified intercloud interoperability conceptual model for IaaS cloud service. In : Model-Driven Engineering and Software Development (MODELSWARD), 2014 2nd International Conference on. IEEE, 2014. p. 673-681.

[19] BERAN, Peter Paul, VINEK, Elisabeth, et SCHIKUTA, Erich. A cloud-based framework for QoS-aware service selection optimization. In : Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services. ACM, 2011. p. 284-287.

[20] SUNDARESWARAN, Smitha, SQUICCIARINI, Anna, et LIN, Dongyang. A brokerage-based approach for cloud service selection. In : Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on. IEEE, 2012. p. 558-565.

[21] MELL, Peter et GRANCE, Timothy. The NIST definition of cloud computing [Recommendations of the National Institute of Standards and Technology-Special Publication 800-145]. Washington DC: NIST. Recuperado de http://csrc. nist. gov/publications/nistpubs/800-145/SP800-145. pdf, 2011.

[22] BAUER, Eric et ADAMS, Randee. Reliability and availability of cloud computing. John Wiley and Sons, 2012.

[23] BUYYA, Rajkumar, BROBERG, James, et GOSCINSKI, Andrzej M. (ed.). Cloud computing: principles and paradigms. John Wiley and Sons, 2010.

[24] PAWAR, Archana, SCHOLAR, M. T., et KAPGATE, P. D. A Review on Virtual Machine Scheduling in Cloud Computing. vol, 2014, vol. 3, p. 928-933.

[25] SAFAVIAN, S. Rasoul et LANDGREBE, David. A survey of decision tree classifier methodology. 1990.

[26] QUINLAN, J.. Ross . Induction of decision trees. Machine learning, 1986, vol. 1, no 1, p. 81-106.

[27] JANIKOW, Cezary Z. Fuzzy decision trees: issues and methods. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 1998, vol. 28, no 1, p. 1-14.

[28] JIN, Chen, DE-LIN, Luo, et FEN-XIANG, Mu. An improved ID3 decision tree algorithm. In : Computer Science and Education, 2009. ICCSE'09. 4th International Conference on. IEEE, 2009. p. 127-130.

[29] RISH, Irina. An empirical study of the naive Bayes classifier. In : IJCAI 2001 workshop on empirical methods in artificial intelligence. IBM New York, 2001. p. 41-46.

[30] CHEESEMAN, Peter, KELLY, James, SELF, Matthew, et al. Autoclass: A Bayesian classification system. In : Readings in knowledge acquisition and learning. Morgan Kaufmann Publishers Inc., 1993. p. 431-441.

[31] MURPHY, Kevin P. Naive bayes classifiers. University of British Columbia, 2006.

[32] ZHANG, Harry. The optimality of naive Bayes. AA, 2004, vol. 1, no 2, p. 3.

[33] KIBRIYA, Ashraf M., FRANK, Eibe, PFAHRINGER, Bernhard, et al. Multinomial naive bayes for text categorization revisited. In : AI 2004: Advances in Artificial Intelligence. Springer Berlin Heidelberg, 2004. p. 488-499.

[34] HIPP, Jochen, GNTZER, Ulrich, et NAKHAEIZADEH, Gholamreza. Algorithms for association rule mininga general survey and comparison. ACM sigkdd explorations newsletter, 2000, vol. 2, no 1, p. 58-64.

[35] AGRAWAL, Rakesh, IMIELINSKI, Tomasz, et SWAMI, Arun. Mining association rules between sets of items in large databases. ACM SIGMOD Record, 1993, vol. 22, no 2, p. 207-216.

[36] MA, Bing Liu Wynne Hsu Yiming. Integrating classification and association rule mining. In : Proceedings of the fourth international conference on knowledge discovery and data mining. 1998.

[37] PADHY, Neelamadhab et PANIGRAHI, Rasmita. Multi Relational Data Mining Approaches: A Data Mining Technique. arXiv preprint arXiv:1211.3871, 2012.

[38] HEARST, Marti A.. , DUMAIS, Susan T., OSMAN, Edgar, et al. Support vector machines. Intelligent Systems and their Applications, IEEE, 1998, vol. 13, no 4, p. 18-28.

[39] KLEIN, Adrian, ISHIKAWA, Fuyuki, et HONIDEN, Shinichi. Efficient heuristic approach with improved time complexity for qos-aware service composition. In : Web Services (ICWS), 2011 IEEE International Conference on. IEEE, 2011. p. 436-443.

[40] SRINIVASAN, S. (ed.). Security, Trust, and Regulatory Aspects of Cloud Computing in Business Environments. IGI Global, 2014.

[41] HABIB, Sheikh Mahbub, RIES, Sebastian, et MHLH?USER, Max. Cloud computing landscape and research challenges regarding trust and reputation. In : Ubiquitous Intelligence and Computing and 7th International Conference on Autonomic and Trusted Computing (UIC/ATC), 2010 7th International Conference on. IEEE, 2010. p. 410-415.

[42] BUYYA, Rajkumar, YEO, Chee Shin, VENUGOPAL, Srikumar, et al. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation computer systems, 2009, vol. 25, no 6, p. 599-616.

[43] J. Burt. Gartner. Predicts Rise of Cloud Service Brokerages. http://www.eweek.com/c/a/Cloud-Computing/GartnerPredict-Rise-of-Cloud-Service-Brokerages-759833/.

[44] SMITH, D. M. Cloud services brokerages: the dawn of the next intermediation age. Cloud Services Brokerage. Gartner. com, 2012.

[45] MONDAL, Anirban, YADAV, Kuldeep, et MADRIA, Sanjay Kumar. EcoBroker: An economic incentive-based brokerage model for efficiently handling multiple-item queries to improve data availability via replication in mobile-p2p networks. In : Databases in Networked Information Systems. Springer Berlin Heidelberg, 2010. p. 274-283.

[46] TAYLOR, Stuart, YOUNG, Andy, et MACAULAY, James. Small Businesses Ride the Cloud: SMB Cloud Watch-US Survey Results. Cisco Internet Business Solutions Group, 2010, p. 1-13.

[47] JULA, Amin, SUNDARARAJAN, Elankovan, et OTHMAN, Zalinda. Cloud computing service composition: A systematic literature review. Expert Systems with Applications, 2014, vol. 41, no 8, p. 3809-3824.

[48] ZISSIS, Dimitrios et LEKKAS, Dimitrios. Addressing cloud computing security issues. Future Generation computer systems, 2012, vol. 28, no 3, p. 583-592

[49] GUTIERREZ-GARCIA, J. Octavio et SIM, Kwang Mong. Agent-based cloud service composition. Applied intelligence, 2013, vol. 38, no 3, p. 436-464.

[50] WEI, Yi et BLAKE, M. Brian. Service-oriented computing and cloud computing: challenges and opportunities. IEEE Internet Computing, 2010, no 6, p. 72-75.

[51] STRUNK, Anja. QoS-aware service composition: A survey. In : Web Services (ECOWS), 2010 IEEE 8th European Conference on. IEEE, 2010. p. 67-74.

[52] HUO, Ying, ZHUANG, Yi, GU, Jingjing, et al. Discrete gbest-guided artificial bee colony algorithm for cloud service composition. Applied Intelligence, 2015, vol. 42, no 4, p. 661-678.

[53] MIN, Xunyou, XU, Xiaofei, et WANG, Zhongjie. Combining Von Neumann Neighborhood Topology with Approximate-Mapping Local Search for ABC-Based Service Composition. In : Services Computing (SCC), 2014 IEEE International Conference on. IEEE, 2014. p. 187-194.

[54] KRITIKOS, Kyriakos et PLEXOUSAKIS, Dimitris. Multi-Cloud Application Design through Cloud Service Composition. In : Cloud Computing (CLOUD), 2015 IEEE 8th International Conference on. IEEE, 2015. p. 686-693.

[55] KRITIKOS, Kyriakos et PLEXOUSAKIS, Dimitris. Multi-Cloud Application Design through Cloud Service Composition. In : Cloud Computing (CLOUD), 2015 IEEE 8th International Conference on. IEEE, 2015. p. 686-693.

[56] ZOU, Guobing, CHEN, Y., YANG, Y., et al. AI planning and combinatorial optimization for web service composition in cloud computing. In : Proc international conference on cloud computing and virtualization. 2010. p. 1-8.

[57] WANG, Xianzhi, WANG, Zhongjie, et XU, Xiaofei. An Improved Artificial Bee Colony Approach to QoS-Aware Service Selection. In : Web Services (ICWS), 2013 IEEE 20th International Conference on. IEEE, 2013. p. 395-402.

[58] ALRIFAI, Mohammad et RISSE, Thomas. Combining global optimization with local selection for efficient QoS-aware service composition. In : Proceedings of the 18th international conference on World wide web. ACM, 2009. p. 881-890.

[59] ZENG, Liangzhao, BENATALLAH, Boualem, NGU, Anne HH, et al. Qos-aware middleware for web services composition. Software Engineering, IEEE Transactions on, 2004, vol. 30, no 5, p. 311-327.

[60] JIN, Hong, YAO, Xifan, et CHEN, Yong. Correlation-aware QoS modeling and manufacturing cloud service composition. Journal of Intelligent Manufacturing, 2015, p. 1-14.

[61] KURDI, Heba, AL-ANAZI, Abeer, CAMPBELL, Carlene, et al. A combinatorial optimization algorithm for multiple cloud service composition. Computers and Electrical Engineering, 2015, vol. 42, p. 107-113.

[62] DOU, Wanchun, ZHANG, Xuyun, LIU, Jianxun, et al. HireSome-II: Towards privacy-aware cross-cloud service composition for big data applications. Parallel and Distributed Systems, IEEE Transactions on, 2015, vol. 26, no 2, p. 455-466.

[63] HUANG, Biqing, LI, Chenghai, et TAO, Fei. A chaos control optimal algorithm for QoS-based service composition selection in cloud manufacturing system. Enterprise Information Systems, 2014, vol. 8, no 4, p. 445-463.

[64] KARIM, Raed, DING, Chen, et MIRI, Ali. End-to-End QoS Prediction of Vertical Service Composition in the Cloud. In : Cloud Computing (CLOUD), 2015 IEEE 8th International Conference on. IEEE, 2015. p. 229-236.

[65] CANFORA, Gerardo, DI PENTA, Massimiliano, ESPOSITO, Raffaele, et al. An approach for QoS-aware service composition based on genetic algorithms. In : Proceedings of the 7th annual conference on Genetic and evolutionary computation. ACM, 2005. p. 1069-1075.

[66] YILMAZ, Ali E. et KARAGOZ, Pinar. Improved Genetic Algorithm Based Approach for QoS Aware Web Service Composition. In : Web Services (ICWS), 2014 IEEE International Conference on. IEEE, 2014. p. 463-470.

[67] LIU, Huan, ZHONG, Farong, OUYANG, Bang, et al. An approach for qos-aware web service composition based on improved genetic algorithm. In : Web Information Systems and Mining (WISM), 2010 International Conference on. IEEE, 2010. p. 123-128.

[68] KLEIN, Adrian, ISHIKAWA, Fuyuki, et HONIDEN, Shinichi. Efficient heuristic approach with improved time complexity for qos-aware service composition. In : Web Services (ICWS), 2011 IEEE International Conference on. IEEE, 2011. p. 436-443.

[69] LI, Minghui, WU, Kaigui, et LIU, Lu. QoS-aware service composition in multi-network environment based on genetic algorithm. In : Communications and Networking in China (CHINACOM), 2011 6th International ICST Conference on. IEEE, 2011. p. 1231-1235.

[70] KLEIN, Adrian, WAGNER, Florian, ISHIKAWA, Fuyuki, et al. A Probabilistic Approach for Long-Term B2B Service Compositions. In : Web Services (ICWS), 2012 IEEE 19th International Conference on. IEEE, 2012. p. 259-266.

[71] WU, Huijun et HUANG, Dijiang. Mosec: Mobile-cloud service composition. In : 3rd international conference on mobile cloud computing, services, and engineering (MobileCloud). IEEE. 2015.

[72] BAO, Huihui et DOU, Wanchun. A QoS-aware service selection method for cloud service composition. In : Parallel and Distributed Processing Symposium Workshops and PhD Forum (IPDPSW), 2012 IEEE 26th International. IEEE, 2012. p. 2254-2261.

[73] GUTIERREZ-GARCIA, J. Octavio et SIM, Kwang-Mong. Self-organizing agents for service composition in cloud computing. In : Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on. IEEE, 2010. p. 59-66.

[74] YU, Qi et BOUGUETTAYA, Athman. Efficient service skyline computation for composite service selection. Knowledge and Data Engineering, IEEE Transactions on, 2013, vol. 25, no 4, p. 776-789.

[75] JULA, Amin, OTHMAN, Zulkifli, et SUNDARARAJAN, Elankovan. A hybrid imperialist competitive-gravitational attraction search algorithm to optimize cloud service composition. In : Memetic Computing (MC), 2013 IEEE Workshop on. IEEE, 2013. p. 37-43.

[76] BADIDI, Elarbi. A cloud service broker for SLA-based SaaS provisioning. In : Information Society (i-Society), 2013 International Conference on. IEEE, 2013. p. 61-66.

[77] WU, Quanwang, ZHU, Qingsheng, et ZHOU, Mingqiang. A correlation-driven optimal service selection approach for virtual enterprise establishment. Journal of Intelligent Manufacturing, 2014, vol. 25, no 6, p. 1441-1453.

[78] GARG, Saurabh Kumar, VERSTEEG, Steve, et BUYYA, Rajkumar. A framework for ranking of cloud computing services. Future Generation Computer Systems, 2013, vol. 29, no 4, p. 1012-1023.

[79] XU, Hong et LI, Baochun. A general and practical datacenter selection framework for cloud services. In : Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on. IEEE, 2012. p. 9-16.

[80] PEARSON, Siani et SANDER, Tomas. A mechanism for policy-driven selection of service providers in SOA and cloud environments. In : New Technologies of Distributed Systems (NOTERE), 2010 10th Annual International Conference on. IEEE, 2010. p. 333-338.

[81] WU, Quanwang, ZHU, Qingsheng, et LI, Peng. A neural network based reputation bootstrapping approach for service selection. Enterprise Information Systems, 2015, vol. 9, no 7, p. 768-784.

[82] LIU, Ran, YUAN, Xiaoqun, XU, Jie, et al. A novel server selection approach for mobile cloud streaming service. Simulation Modelling Practice and Theory, 2015, vol. 50, p. 72-82.

[83] DING, Zhijun, SUN, Youqing, LIU, Junjun, et al. A genetic algorithm based approach to transactional and QoS-aware service selection. Enterprise Information Systems, 2015, p. 1-20.

[84] RUIZ-ALVAREZ, Arkaitz et HUMPHREY, Marty. An automated approach to cloud storage service selection. In : Proceedings of the 2nd international workshop on Scientific cloud computing. ACM, 2011. p. 39-48.

[85] OLIVEIRA, Tiago, THOMAS, Manoj, et ESPADANAL, Mariana. Assessing the determinants of cloud computing adoption: An analysis of the manufacturing and services sectors. Information and Management, 2014, vol. 51, no 5, p. 497-510.

[86] WANG, Xiaogang, CAO, Jian, et XIANG, Yang. Dynamic cloud service selection using an adaptive learning mechanism in multi-cloud computing. Journal of Systems and Software, 2015, vol. 100, p. 195-210.

[87] LI, Chunlin. Hybrid cloud service selection strategy: Model and application of campus. Computer Applications in Engineering Education, 2015.

[88] Zhang, Miranda, et al. "Investigating decision support techniques for automating cloud service selection." Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on. IEEE, 2012.

[89] Ghezzi, Carlo, et al. "Performance-driven dynamic service selection." Concurrency and Computation: Practice and Experience 27.3 (2015): 633-650.

[90] Mohammed, Merzoug, Mohammed Amine Chikh, and Hadjila Fethallah. "QoS-aware web service selection based on harmony search." ISKO-Maghreb: Concepts and Tools for knowledge Management (ISKO-Maghreb), 2014 4th International Symposium. IEEE, 2014.

[91] Skoutas, Dimitrios, et al. "Ranking and clustering web services using multicriteria dominance relationships." Services Computing, IEEE Transactions on 3.3 (2010): 163-177.

[92] He, Qiang, et al. "Quality-aware service selection for service-based systems based on iterative multi-attribute combinatorial auction." Software Engineering, IEEE Transactions on 40.2 (2014): 192-215.

[93] Wang, Shangguang, et al. "Cloud model for service selection." Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on. IEEE, 2011.

[94] JOHN HENRY HOLLAND. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press, 1992.

[95] SAATY, Thomas L. Decision making for leaders: the analytic hierarchy process for decisions in a complex world. RWS publications, 1990.

[96] Garg S. K., Versteeg S., and Buyya R. (2011, December). Smicloud: A framework for comparing and ranking cloud services. In Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on (pp. 210-218). IEEE.

[97] Buyukyazlcl M., and Sucu M. (2003). The analytic hierarchy and analytic network processes. CRITERION, 1, C1.

[98] Saaty T. L. (1990). How to make a decision: the analytic hierarchy process. European journal of operational research, 48(1), 9-26.

[99] Godse M., and Mulik S. (2009, September). An approach for selecting software-as-a-service (SaaS) product. In Cloud Computing, 2009. CLOUD'09. IEEE International Conference on (pp. 155-158). IEEE.

[100] Boussoualim N., and Aklouf Y. (2014, April). An Approach based on user preferences for selecting SaaS product. In Multimedia Computing and Systems (ICMCS), 2014 International Conference on (pp. 1182-1188). IEEE.

[101] Karim R. Chen Ding, Miri A. An end-to-end Qos mapping approach for cloud service selection. In: Proceedings of the IEEE 9th world congress on services (SERVICES). Santa Clara Marriott, CA; pp.341-348, 2013.

[102] Nie Guihua, Qiping She, and Donglin Chen. Evaluation Index System of Cloud Service and the Purchase Decision-Making Process Based on AHP. Proceedings of the 2011 International Conference on Informatics, Cybernetics, and Computer

Engineering (ICCE2011) November 19-20, 2011, Melbourne, Australia. Springer Berlin Heidelberg, pp. 345-352, 2012.

[103] Han S. M., Hassan M. M., Yoon C. W., and Huh, E. N. (2009, November). Efficient service recommendation system for cloud computing market. In Proceedings of the 2nd international conference on interaction sciences: information technology, culture and human (pp. 839-845). ACM.

[104] Limam N., and Boutaba R. (2010). Assessing software service quality and trustworthiness at selection time. Software Engineering, IEEE Transactions on, 36(4), 559-574.

[105] Saripalli P., and Pingali G. (2011, July). Madmac: Multiple attribute decision methodology for adoption of clouds. In Cloud Computing (CLOUD), 2011 IEEE International Conference on (pp. 316-323). IEEE.

[106] W. Wu. Mining significant factors affecting the adoption of SaaS using the rough set approach. The journal of systems and software 84, pp. 435-441, 2010.

[107] COPIL, Georgiana, TRIHINAS, Demetris, TRUONG, Hong-Linh, et al. ADVISECa Framework for Evaluating Cloud Service Elasticity Behavior. In : Service-Oriented Computing. Springer Berlin Heidelberg, 2014. p. 275-290.

[108] BALDUZZI, Marco, ZADDACH, Jonas, BALZAROTTI, Davide, et al. A security analysis of amazon's elastic compute cloud service. In : Proceedings of the 27th Annual ACM Symposium on Applied Computing. ACM, 2012. p. 1427-1434.

[109] CROPLEY, David H., CROPLEY, Arthur J., CHIERA, Belinda A., et al. Diagnosing organizational innovation: Measuring the capacity for innovation. Creativity Research Journal, 2013, vol. 25, no 4, p. 388-396.

[110] MARTENS, Benedikt, WALTERBUSCH, Marc, et TEUTEBERG, Frank. Costing of cloud computing services: A total cost of ownership approach. In : System Science (HICSS), 2012 45th Hawaii International Conference on. IEEE, 2012. p. 1563-1572.

[111] KULVATUNYOU, Boonserm, LEE, Yunsu, IVEZIC, Nenad, et al. A framework to canonicalize manufacturing service capability models. Computers and Industrial Engineering, 2015, vol. 83, p. 39-60.

[112] CARROLL, Noel, HELFERT, Markus, et LYNN, Theo. Towards the development of a cloud service capability assessment framework. In : Continued Rise of the Cloud. Springer London, 2014. p. 289-336.

[113] CHRISTENSEN, Clayton et RAYNOR, Michael. The innovator's solution: Creating and sustaining successful growth. Harvard Business Review Press, 2013.

[114] LIPSMAN, Andrew, MUDD, Graham, RICH, Mike, et al. The power of" like": How brands reach (and influence) fans through social-media marketing. Journal of Advertising research, 2012, vol. 52, no 1, p. 40.

[115] PHAM, Michel Tuan, GEUENS, Maggie, et DE PELSMACKER, Patrick. The influence of ad-evoked feelings on brand evaluations: Empirical generalizations from consumer responses to more than 1000 TV commercials. International Journal of Research in Marketing, 2013, vol. 30, no 4, p. 383-394.

[116] Z. Pawlak. Rough sets. International journal of computer and information sciences, pp. 341-356, 1982.

[117] LIU, Yongwen, ESSEGHIR, Moez, et BOULAHIA, Leila Merghem. Cloud service selection based on rough set theory. In : Network of the Future (NOF), 2014 International Conference and Workshop on the. IEEE, 2014. p. 1-6.

[118] A. Skowron, J. komorowski, Z. Pawlak and L. Polkowski. Rough set perspective on data and knowledge. Handbook of data mining and knowledge discovery, Oxford university press, pp. 134-149, 2002.

[119] S. Rissino and G. Lambert-Torres. Rough set theory-fundamental concepts, principals, data extraction and applications. Data mining and knowledge discovery in real lite applications, pp. 438-462, 2009.

[120] Zhao Yuxin. 2014 cloud service providers charts. China Internet Weekly, vol. 24, pp. 62-63, 2014.

[121] UCI Machine Learning Repository: Data sets. https://archive.ics.uci.edu/ml/datasets.html

[122] J. Hurwitz, M. Kaufman, F. Halper and D. Kirsch. Hybrid Cloud For Dummies. Wiley, 2012

[123] Z. Pawlak.Rough sets. International journal of computer and information sciences, pp. 341-356, 1982

[124] A. Skowron, J. komorowski, Z. Pawlak and L. Polkowski. Rough set perspective on data and knowledge. Handbook of data mining and knowledge discovery, Oxford university press, pp 134-149, 2002

[125] S. Rissino and G. Lambert-Torres. Rough set theory-fundamental concepts, principals, data extraction, and applications. Data mining and knowledge discovery in real lite applications, pp. 438-462, 2009

[126] C. Y. Mao. Rough set-based debugging for web services system. IEEE Asia-Pacific service Computing Conference, pp. 293-299, 2010

[127] L. F. Ai and M. L. Tang. QoS-based web service composition accommodating inter-service dependencies using minimal-conflict hill-climbing repair genetic algorithm. IEEE Fourth International conference on e-Science, pp. 119-126, 2008

[128] L. F. Ai and M. L. Tang. A penalty-based genetic algorithm for QoS-aware web service composition with inter-service dependencies and conflicts. International Conference on Computational Intelligence for Modeling, Control and Automation, pp. 738-743, 2008

[129] M. Aiello, E. El Khoury, A. Lazovik and P. Ratelband. Optimal QoS-aware web service composition. International Conference on E-Commerce Technology, pp. 491-494, 2009

[130] M. L. Tang and L. F. Ai. A hybrid genetic algorithm for the optimal constrained web service selection problem in web service composition. Evolutionary Computation (CEC), IEEE, pp. 1-8, 2010

[131] Q. Fang, X. Peng, Q. Liu, and Y.Hu. A Global QoS optimizing web services selection algorithm based on MOACO for dynamic web service Composition. International Forum on Information Technology and Applications, pp. 37-42, 2009

[132] A. Huang, C. Lan and S. Yang. An optimal Qos-based web service selection scheme. Information Science: an International Journal, Vol. 179, pp. 3309-3322, 2009

[133] Zia ur Rehman, Omar K. Hussain, Sazia Parvin and Farook K. Hussain. A Framework for User feedback Based Cloud service Monitoring. Sixth International Conference on Complex, Intelligent, and Software Intensive Systems, pp. 257-262, 2012

[134] W. Y. Zeng, Y. L. Zhao and J. W. Zeng. Cloud service and service selection algorithm research. GEC 09, ACM, pp. 1045-1048. 2009

[135] L. Qu, Y. Wang and Mehmet A. Orgun. Cloud Service Selection Based on the Aggregation of User Feedback and Quantitative Performance Assessment. 10th International Conference on Service Computing, pp. 152-159, 2013

[136] Zia ur Rehman, Omar K. Hussain and Farook K. Hussain. Towards multi-criteria cloud service selection. Innovative mobile and internet services in ubiquitous computing, pp. 44-48, 2011

Yongwen LIU

Doctorat : Ingénierie Sociotechnique des Connaissances, des Réseaux et du Développement Durable

Année 2016

Sélection de service cloud en utilisant la théorie des ensembles approximatifs

Avec le développement du cloud computing, de nouveaux services voient le jour et il devient primordial que les utilisateurs aient les outils nécessaires pour choisir parmi ses services.
La théorie des ensembles approximatifs représente un bon outil de traitement de données incertaines. Elle peut exploiter les connaissances cachées ou appliquer des règles sur des ensembles de données. Le but principal de cette thèse est d'utiliser la théorie des ensembles approximatifs pour aider les utilisateurs de cloud computing à prendre des décisions. Dans ce travail, nous avons, d'une part, proposé un cadre utilisant la théorie des ensembles approximatifs pour la sélection de services cloud et nous avons donné un exemple en utilisant les ensembles approximatifs dans la sélection de services cloud pour illustrer la pratique et analyser la faisabilité de cette approche. Deuxièmement, l'approche proposée de sélection des services cloud permet d'évaluer l'importance des paramètres en fonction des préférences de l'utilisateur à l'aide de la théorie des ensembles approximatifs. Enfin, nous avons effectué des validations par simulation de l'algorithme proposé sur des données à large échelle pour vérifier la faisabilité de notre approche en pratique.
Les résultats de notre travail peuvent aider les utilisateurs de services cloud à prendre la bonne décision et aider également les fournisseurs de services cloud pour cibler les améliorations à apporter aux services qu'ils proposent dans le cadre du cloud computing.

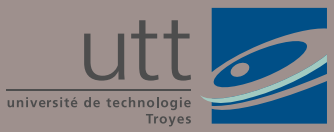
Mots clés : théorie des ensembles approximatifs - prise de décision - informatique dans les nuages - systèmes d'aide à la décision - classification - services web.

Cloud Services Selection based on Rough Set Theory

With the development of the cloud computing technique, users enjoy various benefits that high technology services bring. However, there are more and more cloud service programs emerging. So it is important for users to choose the right cloud service. For cloud service providers, it is also important to improve the cloud services they provide, in order to get more customers and expand the scale of their cloud services.
Rough set theory is a good data processing tool to deal with uncertain information. It can mine the hidden knowledge or rules on data sets. The main purpose of this thesis is to apply rough set theory to help cloud users make decision about cloud services. In this work, firstly, a framework using the rough set theory in cloud service selection is proposed, and we give an example using rough set in cloud services selection to illustrate and analyze the feasibility of our approach. Secondly, the proposed cloud services selection approach has been used to evaluate parameters importance based on the users' preferences. Finally, we perform experiments on large scale dataset to verity the feasibility of our proposal.
The performance results can help cloud service users to make the right decision and help cloud service providers to target the improvement about their cloud services.

Keywords: rough sets - decision making - cloud computing - decision support systems - classification - web services.

Thèse réalisée en partenariat entre :



Ecole Doctorale "Sciences et Technologies"