CLASSIFICATION SYSTEM FOR TB DISEASE USING MOBILE AGENT

Moh Moh Khaing [1], Sai Sie Thu Kyaw[2], Dr. Thae Nu Nge [3]

Technological University, Taunggyi, Shan State, Myanmar, India

Abstract

Data mining is the process of analysing data from different perspectives and summarizing it into useful information. It uses a variety of data analysis tools to discover patterns and relationships in data that may be used to make valid predictions. This system intends to use one of the data mining techniques called Naïve Bayesian Classification to classify TB diseases based on the symptoms of patients. Mobile Agents (MAs) is a relatively new emerging design paradigm for distributed applications over the past few years and they promise to alleviate many of the shortcomings that address in the client-server approach. This is an idea to study this new design paradigm and use it as an alternative implementation mechanism for distributed system. And therefore, we implement this system as a distributed data mining system using mobile agent to allow the distance patients to test their TB diseases using their symptoms.

Keyword: Mobile Agents, Naïve Bayesian, TB.

1.INTRODUCTION

The process of data analysis in health care is becoming more and more complicated for a number of reasons:

- New techniques, such as micro arrays have been rise to the generation of data with unusual characteristics, such as where a few patients are described by a large number of variables.
- The integrated analysis of data from different sources concerning the same health-care topic, including the wish to incorporate background knowledge in the analysis process, is becoming more and more relevant.
- With the widely availability of sophisticated and cheap computing equipment, the exploitation of models to support clinical decision making has become a practical option.

Gradually increasing trend in the health-care field is the use of Bayesian statistical methods in data analysis. Data analysis is then viewed as the process of updating prior knowledge based on available biomedical and health-care evidence in the form of data. In this system, Naive Bayesian Classification is used to classify TB diseases based on the symptoms of patients.

Mobile Agents (MAs), a relatively new emerging design paradigm for distributed applications over the past few years. MAs are bringing together telecommunications, software and distributed system technologies to create new ways of building computing systems. This is an idea to study this new design paradigm and use it as an alternative implementation mechanism for distributed system. And therefore, we implement this system as a distributed data mining system using mobile agent.

2. AIM AND OBJECTIVES

This system intends to implement as the distributed mobile agent application for data mining of TB diseases classification. This research is done to investigate the use of mobile agent in distributed environment for using data mining application. The major objectives of the research are as follows:

- To know why mobile agents are needed in distributed system
- To present an architecture of mobile agent for distributed information retrieval
- To provide necessary data discovery mechanisms, which allow the user to find data based on characteristics of the data.
- To develop the system which can provide necessary data in efficient way and provide location

transparency, that is, users need not to know where the data locates.

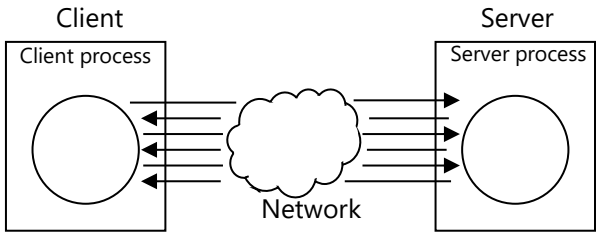
- To understand how to implement the user friendly classification system

To understand how to extract models describing important data classes by using Naive Bayesian Classification. The process of data analysis in health care is becoming more and more complicated for a number of reasons:

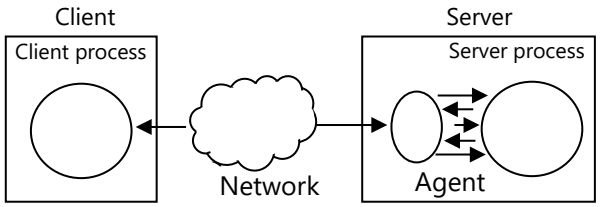
3.BACKGROUND THEORY

3.1. Remote Procedure Call (RPC) verses Mobile Agent Approach

The original motivation behind mobile agents is to replace remote procedure call (RPC) as a way of processes to communicate over a network as shown in figure 1. With RPC, one process can invoke a procedure (method) on another process which is remotely located. In RPC, communication is 'synchronous.' When the network fails, the clients may remain indefinitely suspended, waiting for a reply that will never come.



1(a)



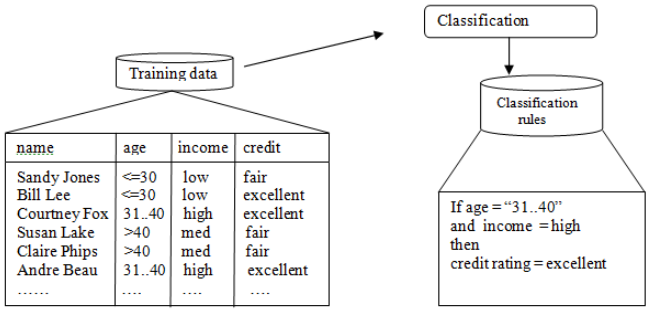
1(b)

Figure 1(a) Remote procedure call (RPC) (b) Mobile agent approach

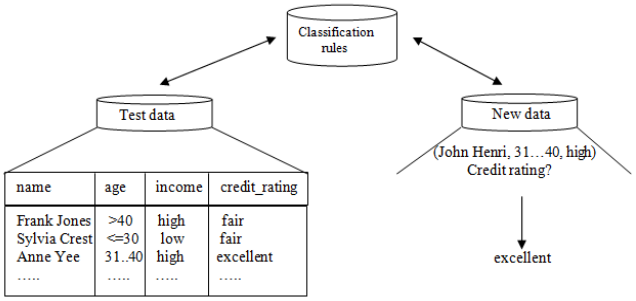
The network connection may remain open and even through it is largely unused (no data is being sent for most of the time), this may be costly. The idea of mobile agent is to replace the RPC by sending out an agent to do computation. Thus instead of invoking a method, a process at client sends out a program, a mobile agent, to a process at server. This program then interacts with server's process. When the agent has completed its interactions, it returns to client's process with the required result. During the entire operation, the only network time required is that to send the agent to server and that required going back the agent to client when it has completed its tasks. This is potentially a much more efficient use of network resources than the RPC.

3.2. Classification

Classification is the process of finding a set of modules (or functions) that describes and distinguish data classes or concepts. Data classification is a two-step process as shown in figure 2.



2(a)



2(b)

Figure 2 The data classification process (a) learning 2(b) classification

Training data are analysed by a classification algorithm. The learned model or classifier is represented in the form of classification rules. Test data are used to estimate the accuracy of the classification rules. If the

accuracy is considered acceptable, the rules can be applied to the classification of new data tuples.
There are three types of classification: eager learners, lazy learners and other classification methods. The followings are the techniques of classification of eager learners. They, when given a set of training tuples, will construct a generalization (i.e., classification) model before receiving new (e.g., test) tuples to classify.

- Decision Tree Induction:
- Bayesian Classification
- Rule-Based Classification:
- Back propagation:
- Support Vector Machine

This system is based on Naïve Bayesian Classifier. Naïve Bayes is an effective and efficient learning algorithm in classification. A Naïve Bayesian classifier is a simple probabilistic classifier based on Bayes' theorem. It predicts class membership probabilities, such as the probability that a given sample belongs to a particular class. It assumes that the effect of an attribute value on a given class is independent of the values of the other attributes. This is called the class conditional independence. It works much better in many complex real-world situations than one might expect. It requires a small amount of training data to estimate the parameters necessary for classification.

4. SYSTEM DESIGN AND IMPLEMENTATION

This section presents the architecture that can build a distributed system using mobile agent to find the classification result for Tuberculosis (TB) disease.

4.1. TB Disease Classification

In this system, the input is a set of training samples and attributes of new patient and comes from the user or client using mobile agent carrying patient's attributes. From these inputs, the probability of each class is computed by using Bayesian Theorem. The algorithm of Bayesian method used in this system. Then the maximum probability value is calculated to define the class of the patient's TB stage.
This system will classify TB diseases for three categories:

- P (Pulmonary Tuberculosis)
- EP (Extra Pulmonary Tuberculosis) and
- No-TB (class for non TB-suffering patients).

This system uses the training data and testing data for both classifying rules and testing the accuracy of the system by using holdout method.
There are 13 attributes in this system. The attributes and values are describes in table 1.

Attributes	**Values**
Age	any
Duration of illness(over 3 weeks)	yes/no
Expectoration	yes/no
Haemoptysis	yes/no
Chest pain	yes/no
Brathlessness	yes/no
Weight Loss	yes/no
Low grade fever	yes/no
Night sweating	yes/no
Loss of appetite	yes/no
Malaise	yes/no
Easy Fatigability	yes/no
Tuberculous intrathoracic lymph	yes/no
Irritability	yes/no
Pre-treatment smear(sputum examination)	3+/2+/1+/neg
Medical Officer's Judgement Family History of Tuberculosis (TB) diseases	yes/no yes/no
Irritability	yes/no

Table 1. Attribute and Value of the Classification System

4.2. Mobile Agents Retrieval

The mobile agent's retrieval of the system is shown in figure 3. In this system include database, testing data, training data, client, Naïve Bayesian classifier and classifier accuracy process are also include.

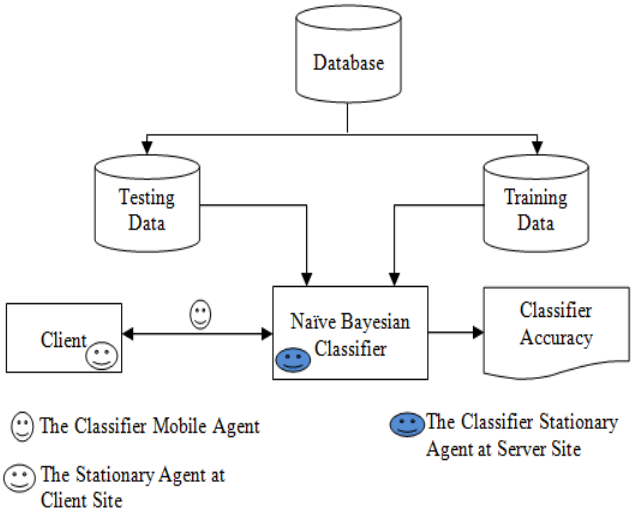


Figure 3 Mobile agents retrieval of the system

Agent Name	Responsibility
Client Stationary Agent	-To accept the name, age and symptoms of patient -To create Classifier Mobile Agent to get classifier result from the sever -To display the classifier result of the patient
Classifier Mobile Agent	-To go to the sever carrying patient's symptoms -To negotiate with Sever Stationary Agent to get the classifier result -To go back to the client with classifier result
Server Stationary Agent	-To apply the classifying process using the patient's symptoms from Classifier Mobile Agent -To give the classifier result to the Classifier Mobile Agent to go back to the client

Table 2. The agent used in this system and their responsibilities

This system uses three agents: one is mobile agent and the other two are stationary agents. The agents used in this system and their responsibilities are shown in table 2.

The components interaction of this system is as shown in figure 4. As shown in this figure, the Classifier Main Agent will be created when a request from the user accepts. Then this agent will accept the patient's information such as name and age of the patient. After that, the symptoms of the patient are also accepted. After getting all the required information, this agent will create the Classifier Mobile Agent to give the information and to go to the server for classifier calculation. As soon as the Classifier Mobile Agent arrives at the server, it collaborates with the Classifier Server Agent. After calculation, the Classifier Server Agent will give the classifier result to the Classifier Mobile Agent. Then the Mobile Agent will come back to the client with the classifier result and then the client will display this result to the user.

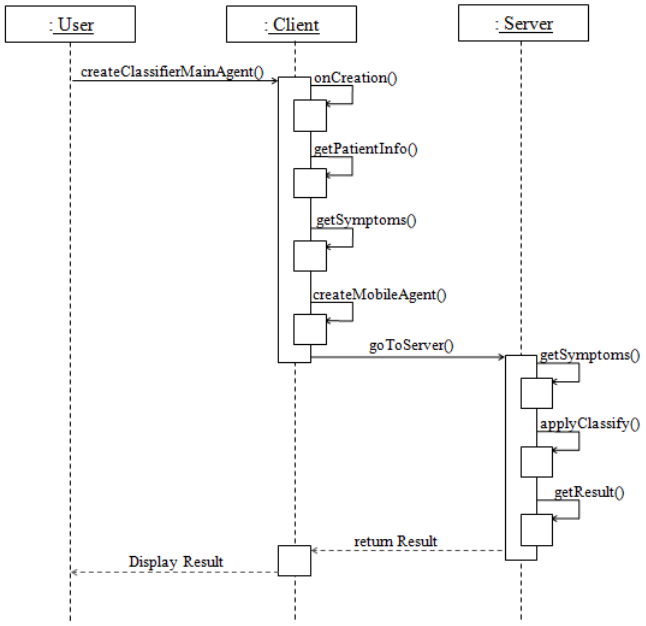


Figure 4 Component interaction of the system

The agent is a one of the Java class. The agents used in this system including mobile agent and stationary agent, are extended or inherited from the Aglet package (the Mobile Agent framework).The relationship between these classes is shown in this figure. At the client site, the 'AcceptInfo', 'AcceptSymptoms' and 'DisplayResult'

classes have relationship with the 'ClientAgent' class. The relationships between them are one-to-one relationships. The class diagram of this system is also shown in figure 5.

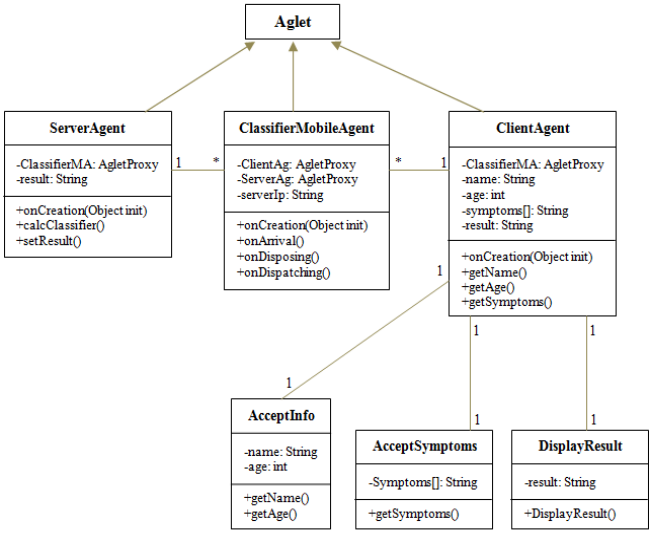


Figure 5 The class diagram of the system

4.3. Classifier Accuracy

Estimating classifier accuracy is important in that it allows one to evaluate how accurately a given classifier will label future data that is data on which the classification has not been trained. Classifier accuracy can be compared with respects to their speed, robustness, scalability, and interpretability. For example, a classifier is trained to classify medical data samples as either "TB" or "No-TB". To evaluate how well it can recognize "TB" samples (positive samples) and to evaluate how well it can recognize "No-TB" samples (negative samples), sensitivity and specificity measures can be used, respectively. Sensitivity and specificity are the most widely used statistics to describe a diagnostic test. Sensitivity is the probability of a positive test among patients with TB disease. Specificity is the probability of a negative test among patients with TB disease. Sensitivity (1) and specificity (2) are calculated by using the following equations. The accuracy (3) is a function of sensitivity and specificity.

$$\text{sensitivity} = \frac{\text{no of true positive}}{\text{no of positive}} = \frac{t - pos}{pos} \qquad (1)$$

$$\text{specificity} = \frac{\text{no of true negative}}{\text{no of negative}} = \frac{t - neg}{neg} \qquad (2)$$

$$\text{accuracy} = \text{sensitivity}\frac{pos}{(pos + neg)} + \text{specificity}\frac{neg}{(pos + neg)} \qquad (3)$$

Where, the term 'pos' is the number of positive samples (presence of TB disease in testing data), 'neg' is the number of negative samples (absence of TB disease in testing data), 't_pos' is the number of true positive samples that will correctly classified by classifier and 't_neg' is the number of true negative sample that will correctly classified by classifier. The accuracy calculation of this system is shown in figure 6.

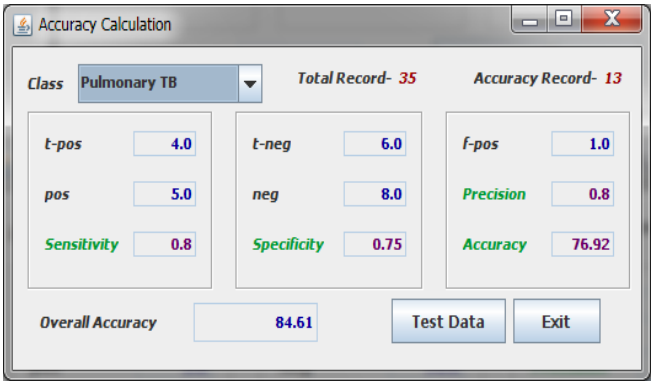


Figure 6 Accuracy calculation of the system

4.4. Design of the System and Implementation

The system consists of two types of user: 'User' part and 'Admin' part. If the user is 'Admin' user, the system asks the password fi and checks it as shown in figure 7. If the password is true, the user can insert, delete or update patient's information; and can calculate the accuracy of the classifier. If the user is 'User', the system asks his symptoms, calculate the class and display the class of his disease.

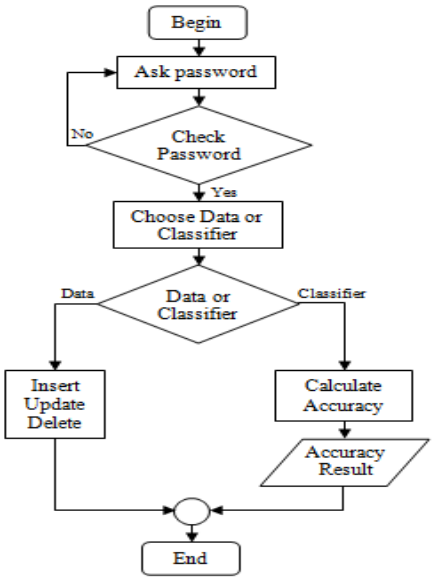


Figure 7 System flow diagram of the admin

The system flow diagram for the user is shown in figure 8.

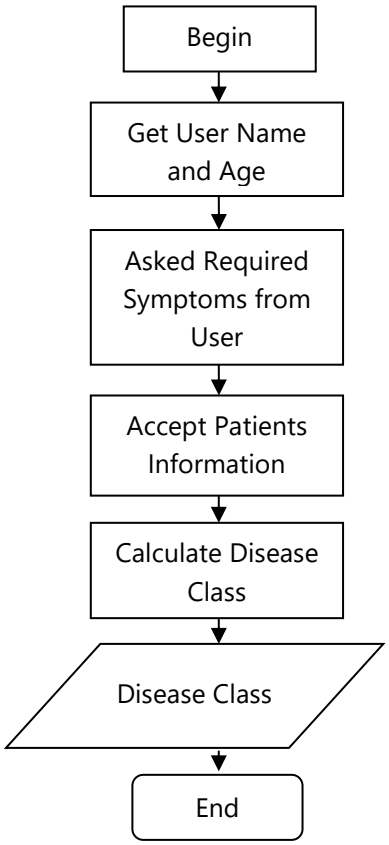


Figure 8 System flow diagram of the user

5. CONCLUSIONS

Most applications of mobile agents center on using the mobile agent as the representative of user, and the mobile agent travels around the network performing tasks on behalf of user. The mobile agent paradigm is much more powerful than this and extremely well suited for designing large-scale applications. Applications whose components have complex changing relationships and are geographically distributed would most benefit from using mobile agent design.

This system is implemented as a by using Naïve Bayesian classification using Mobile agent technology. This system can classify the class of TB diseases based on the symptoms of patients. To measure the accuracy of the system, holdout method is also used in this system. The accuracy of the system depends on the amount of training data; the more the training data, the more the accuracy is. According to the experimental results, the system is reliable for users who would like to know the class of TB disease of someone where the doctors are not available.

REFERENCES

[1] B. W. Lampson, "Distributed System: Architecture and Implementation", Lecture Notes in Computer Science, Springer-Verlag, Berlin.

[2] C. George, D. Jean, and K. Tim, "Distributed Systems – Concepts and Design (Third Edition)", Pearson Education Ltd, Edinburgh Gate, Harlow, Essex CM20 2JE, England.

[3] G. Vigna, "Mobile Agents and Security", Springer- Verlag, 1998.

[4] H.Zhang, L.Jiang and J.Su, "Augmenting Naïve Bayes for Ranking", Proceedings of the 22nd International Conference on Machine Learning, Bonn, Germany, 2005.

[5] J. Han, and M. Kamber, "Data Mining: Concepts and Techniques", CA: Prentice Hall, 2002.

[6] K. Neeran and T. Anand, "Design Issues in Mobile Agent Programming Systems", IEEE Concurrency, July-September 1998.

[7] K. Neeran and T. Anand, "Design Issues in Mobile Agent Programming Systems", IEEE Concurrency, July-September 1998.

[8] T. M. Mitchell, "Generative and Discriminative Classifiers: Naïve Bayes and Logistic

Regression", Machine Learning, McGrawHill, 2005.

[9] W. Michael, "An Introduction to MultiAgent Systems", Department of Computer Science, University of Loverpool, UK.

[10] Y.Tsuruoka, J.Tsujii, "Training a Naïve Bayes Classifier via the EM Algorithm with a Class Distribution Constraint", Department of Computer Science, University of Tokyo.